DHRec: A Debiased Hyperbolic Recommendation Model

Mengmeng Li* (lmm012328@163.com), Hongmei Li (376342117@qq.com),
Zhenyu Li (lizhenyu1165@hotmail.com), and **Xianglong Li (lxlong8901@163.com)**
Academy of Military Sciences
Beijing, 100091 China

Jinlong Tian (tianjinlong@nudt.edu.cn), **Qiyuan Zhang (1115840068@qq.com)**
National University of Defense Technology
Changsha, 410073 China

Abstract

Personalized recommendation aims to recommend candidate items to users based on their preferences by simulating their cognitive decision-making process. User-item interaction data typically follows a power-law distribution. However, existing works usually learn the representations of users and items in Euclidean space, resulting in a mismatch between the data volume space and the embedding space, which causes significant distortion in the representations. Moreover, the presence of cognitive biases, such as conformity, can also introduce distortion in representation learning. Therefore, we propose a Debiased Hyperbolic Recommendation model, called DHRec. Specifically, first, we choose to model the representations of user and item in hyperbolic space, which has exponential growth capabilities. Second, in addition to the user-item interaction graph, we also construct semantic graphs to capture the semantic neighbor information of users and items. Then, by adjusting the weights of neighbor nodes, we learn debiased representations of users and items, effectively alleviating the bias caused by conformity. Finally, we compute the predicted scores between user and candidate items in hyperbolic space. Extensive experiments on three datasets demonstrate that our model surpasses the strongest baseline, achieving a 11.04% and 10.09% improvement on Recall and NDCG, respectively.

Keywords: personalized recommendation; hyperbolic space; conformity bias; debiased representation

Introduction

Recently, with the rapid development of the Internet, the number of items has shown explosive growth, leaving users overwhelmed. Hence, personalized recommendation, which aims to recommend candidate items that the user may be interested in, is necessary for service platforms to help users alleviate information overload as well as improve their shopping experience (S. Wang, Guo, Wang, Liu, & Xu, 2023).

Collaborative filtering (CF)-based methods which focus on mining users' preferences through historical interaction between users and items, have long been one of the essential approaches to achieving efficient personalized recommendations (Li et al., 2023). Early CF-based models mainly used matrix factorization (MF) methods, which decomposed the original user-item interaction matrix into two low-rank matrices to extract latent features of users and items (Koren, Bell, & Volinsky, 2009). However, due to the complex interaction between users and items, the shallow representations in MF-based methods lack the capacity to capture deeper associations between them (Li et al., 2023). With the development of deep learning, people began to use neural network to capture complex user interaction behaviors, which enhanced the performance of recommendation models (Cheng et al., 2016). Due to the graph-structured nature of user-item interactions, graph neural networks (GNNs) have been introduced in recommendation, successfully enhancing the representations through information propagation and aggregation (He et al., 2020b).

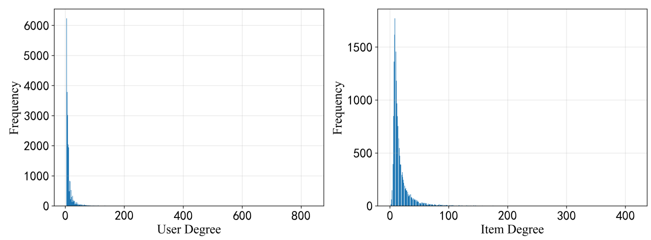

Figure 1: Degree distribution of users and items in Amazon-CD dataset.

We conduct a statistical analysis on the real-world dataset Amazon-CD and find that the user-item interaction graph exhibits a power-law distribution, as shown in Figure 1. (Ravasz & Barabási, 2003) has theoretically proved that this scale-free graph can be traced back to a hierarchical structure and that the volume of the graph grows exponentially with its radius. However, existing recommendation methods usually tend to learn user and item representations in Euclidean space owning polynomial growth volume with respect to radius. For human cognition, this inconsistency between the data volume space and the embedding space will lead to serious representation distortion. In contrast to Euclidean space, hyperbolic space has an area that is exponential with radius, which provides a nice alternative to model underlying hierarchical structure data (Sun, Cheng, Zuberi, Perez, & Volkovs, 2021; M. Yang, Li, Zhou, Liu, & King, 2022). Moreover, the conformity mentality, a common cognitive bias, leads many users to make choices based on social influence, often deviating from their true preferences. This is particularly evident when users interact with popular items, where their choices may not reflect their actual preferences but are driven by social trends (Chen et al., 2023). This conformity bias often distorts the recommendations, as the model tends to favor popular items that may not align with the user's intrinsic preferences. Existing models often overlook this bias, which affects the quality and relevance of recommendations.

In D. Barner, N.R. Bramley, A. Ruggeri and C.M. Walker (Eds.), *Proceedings of the 47th Annual Conference of the Cognitive Science Society*

In this paper, we propose a novel debiased hyperbolic recommendation model, called DHRec. It combines the expressiveness of GNNs with the modeling capabilities of hyperbolic space to enhance node representations (Y. Yang et al., 2024). Additionally, it mitigates the cognitive biases, such as conformity, on recommendation. Specifically, we use the hyperbolic model to model users and items. Then, we learn the representations of users and items by building a GNN-based debiased representation learning module. Finally, we calculate the relevance score between target user and candidate item and decide whether to recommend or not.

Our contributions can be summarized as follows:

(1) A hyperbolic recommendation model. We utilize the superiority of hyperbolic space to conduct user and item modeling, in order to solve inconsistent space capacities problem and obtain low distortion representations.

(2) A GNN-based debiased representation learning module. To capture users' intrinsic preferences, we learn the representations of user and item by aggregating the debiased neighborhood information from the interaction graph and the semantic graph.

(3) The empirical results demonstrate the effectiveness of our approach and verify the validity of hyperbolic space as well as the debiased representation learning module.

Related Works

In recent years, recommendation models based on graph representation learning in hyperbolic space have received increasing attention. Specifically, HGCF (Sun et al., 2021) is the first to combine hyperbolic space with GCN for collaborative recommendation and proposes a SkipGCN architecture to alleviate the issues of gradient vanishing and over-smoothing. LGCF (L. Wang, Hu, Wu, & Wang, 2021) proposes a pure hyperbolic GCN in which all operations are performed in the hyperbolic space to reduce the distortion of user and item representations. HRCF (M. Yang, Zhou, Liu, Lian, & King, 2022) designs a geometry-aware hyperbolic regularization method to alleviate the over-smoothing problem. HICF(M. Yang, Li, et al., 2022) proposes an adaptive loss function suitable for hyperbolic space, which greatly improves the recommendation performance of the model. HGCH (Zhang & Wu, 2024) integrates diverse side information into a heterogeneous collaborative graph to improve the recommendation accuracy, and improves training convergence speed through the designed initialization method and negative sampling method. HG-CLSR (Tu, Meng, Qi, Xu, & Zhang, 2024) proposes a second-order sampling hyperbolic contrastive learning method for collaborative recommendation to improve the performance of the model from a self-supervision perspective. However, although the graph representation learning recommendation method based on hyperbolic space shows significant superiority, there is still a huge room for improvement, because most of the existing research ignores the bias introduced by the conformity mentality, which is what our work focus.

Preliminaries

In this section, we first introduce the hyperbolic space relevant to this paper and then present our task formalization.

Hyperbolic Space

In mathematics, hyperbolic space typically refers to a manifold with constant negative curvature. A key property of hyperbolic spaces is their faster expansion compared to Euclidean spaces. Several hyperbolic geometric models commonly used in the recommendation field are: Poincaré model, Lorentz model, and Klein model (Gulcehre et al., 2018). These models are all connected and can be converted into each other. In this paper, we work with the Lorentz model which is found to be more stable for numeric optimization (Nickel & Kiela, 2018).

We recall that a d-dimensional Lorentz model with a constant negative curvature $-1/K(K>0)$ is a Riemannian manifold $\left(\mathbb{H}^{d,K}, g_{\mathcal{L}}\right)$, where $\mathbb{H}^{d,K} = \{\mathbf{x} \in \mathbb{R}^{d+1} : \langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}} = -K, x_0 > 0\}$, $g_{\mathcal{L}} = \eta$ ($\eta = \mathbf{I}_n$ except $\eta_{0,0} = -1$) and $\langle \cdot, \cdot \rangle_{\mathcal{L}}$ is the Lorentzian inner product. Given $\mathbf{x}, \mathbf{y} \in \mathbb{H}^{d,K}$, the Lorentz inner product is given by:

$$< \mathbf{x}, \mathbf{y} >_{\mathcal{L}} = -x_0 y_0 + x_1 y_1 + \cdots + x_d y_d. \quad (1)$$

In addtion, the d-dimensional Euclidean tangent space centered at vector $\mathbf{x} \in \mathbb{H}^{d,K}$ is denoted as $\mathbb{T}_{\mathbf{x}}^{d,K}$, where $\mathbb{T}_{\mathbf{x}}^{d,K} = \left\{\mathbf{v} \in \mathbb{R}^{d+1} : \langle \mathbf{v}, \mathbf{x} \rangle_{\mathcal{L}} = 0\right\}$. The tangent space $\mathbb{T}_{\mathbf{x}}^{d,K}$ can be considered as the first-order approximation of $\mathbb{H}^{d,K}$ around $\mathbf{x}$.

The exponential map and logarithmic map are proposed to connect hyperbolic space and tangent space. Given $\mathbf{x}, \mathbf{y} \in \mathbb{H}^{d,K}$ and $\mathbf{v} \in \mathbb{T}_{\mathbf{x}}^{d,K}$, the exponential map maps $\mathbf{v}$ from tangent space to hyperbolic space:

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

where $\|\mathbf{v}\|_{\mathcal{L}} = \sqrt{< \mathbf{v}, \mathbf{v} >_{\mathcal{L}}}$ is the Lorentzian norm of $\mathbf{v}$. In contrast to the exponential map, the logarithmic map maps the $\mathbf{y}$ of the hyperbolic space to the tangent space:

$$\log_{\mathbf{x}}^{K}(\mathbf{y}) = d_{\mathcal{L}}^{K}(\mathbf{x}, \mathbf{y}) \frac{\mathbf{y} + \frac{1}{K}\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} \mathbf{x}}{\|\mathbf{y} + \frac{1}{K}\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} \mathbf{x}\|}, \quad (3)$$

where $d_{\mathcal{L}}^{K}(\cdot, \cdot)$ is the hyperbolic distance between points $\mathbf{x}, \mathbf{y} \in \mathbb{H}^{d,K}$, which is defined as follows:

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

For simplicity, we fix parameter K and set it to 1, implying that the curvature is -1.

Task Formulation

In a recommendation scenario, the user set and item set are $\mathcal{U} = \{u_1, u_2, \cdots, u_M\}$ and $I = \{i_1, i_2, \cdots, i_N\}$, where M, N denote the number of users and items, respectively. User-item interaction matrix is $\mathbf{X}_{M \times N} = \{x_{ui} | u \in \mathcal{U}, i \in I\}$, in which x_{ui} = 1 means user u once interacted with item i. The task of recommendation is to predict the probability that user u may interact with an item i that he/she has never seen before.

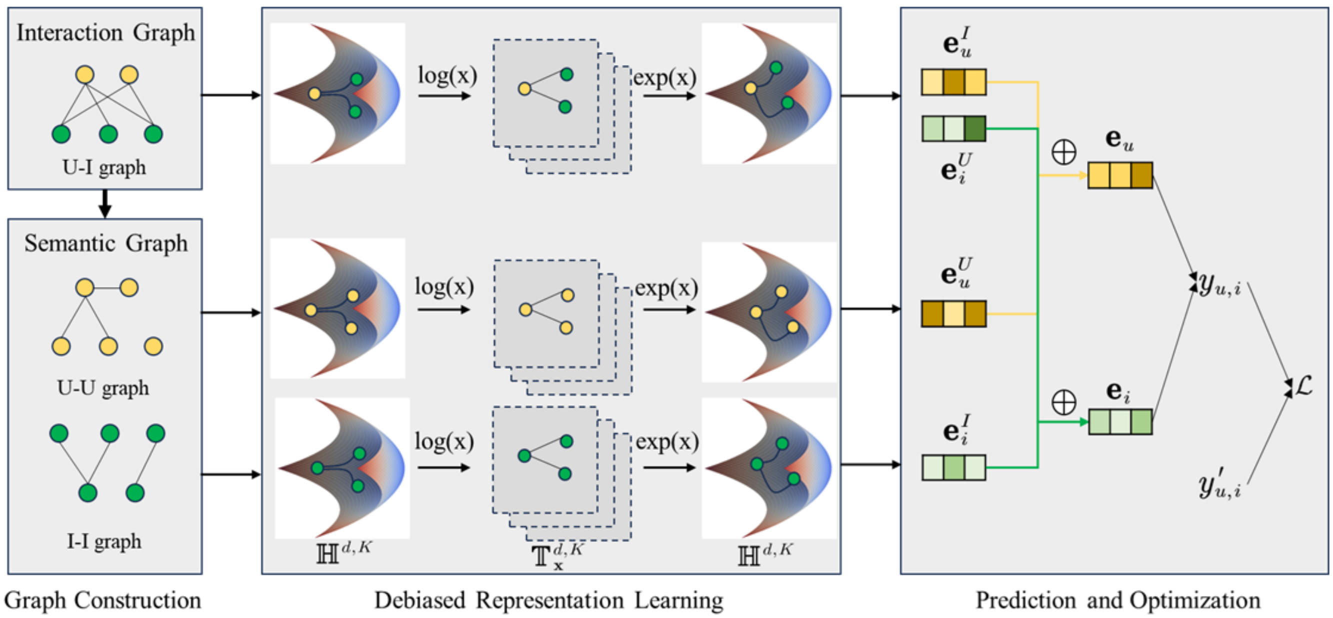

Figure 2: The framework of DHRec. It consists of three parts: graph construction, debiased representation learning, prediction and optimization.

Methodology

In this part, we will introduce our proposed model DHRec in three parts: graph construction, debiased representation learning, prediction and optimization. The framework of DHRec is shown in Figure 2.

Graph Construction

In this subsection, we describe the semantic graph construction process, which includes three steps: (1) constructing the user-item interaction graph, (2) building the user and item relationship graphs, and (3) generating the user and item semantic graphs. We take the generation of the user semantic graph as an example, which also applies to the item semantic graph. Figure 3 illustrates this process of constructing the user semantic graph.

First step, we construct a user-item interaction graph based on user-item historical interaction records.

Second step, we transform the user-item interaction graph to construct the user relationship graph. Specifically, an edge is established between two users if they interacted with one common item before. As shown in Figure 3, user u_a and user u_b have both interacted with item i_a, while user u_a and user u_c have both interacted with item i_c. Therefore, in user relationship graph, there are edges between user u_a and both user u_b and user u_c. In addition, to indicate the strength of the relationship between users, we set a weight w for each edge. Specifically, each weight w consists of two parts: the historical interaction similarity between two users s and the popularity of the co-interaction items p.

We use heat kernel to define the historical interaction similarity s as follows:

$$s_{u_a u_b} = e^{-\frac{\left\|X_{u_a} - X_{u_b}\right\|^2}{t}}, \tag{5}$$

where X_{u_a} and X_{u_b} are the corresponding rows in the user-item interaction matrix X for u_a and u_b. t is the time hyper-parameter. If the historical interaction of u_a and u_b are similar, the factor $s_{u_a u_b}$ should be large, and vice versa.

We measure the popularity of the co-interaction items p by the degree of the co-interaction items. Less popular items are considered to better reflect users' true preferences. Therefore, we define p as follows:

$$p_{u_a u_b} = \frac{2}{|I_{u_a u_b}|} \cdot \sum_{i_j \in I_{u_a u_b}} \frac{1}{|U_{i_j}|}, \tag{6}$$

where $I_{u_a u_b}$ contains the co-interaction items that rated by u_a and u_b and U_{i_j} contains users that rated i_j before.

The weight w is obtained as follows:

$$w_{u_a u_b} = s_{u_a u_b} \cdot p_{u_a u_b}. \tag{7}$$

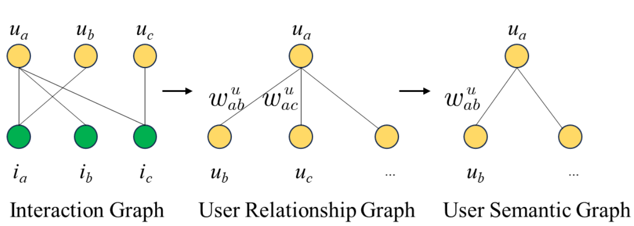

Figure 3: An illustration of user semantic graph construction.

Third step, to reduce hardware overhead and address the potential scale issues of the relational graph, we further construct a semantic graph. Specifically, for each user, we select the top-K neighbors based on the weight w from relationship graph to construct the semantic graph.

Debiased Representation Learning

In this subsection, we will introduce in detail the debiased representation learning based on hyperbolic space and GNNs.

Hyperbolic Embedding Initialization. First, we generate the initial node representations $\mathbf{x} \in \mathbb{R}^d$ (for users and items) in Euclidean space by looking up the table. Then, we insert value 0 at the 0th coordinate of $\mathbf{x}$ so that $\mathbf{z}^0$ can always live in the tangent space of north pole (origin) $\mathbf{o} := \left\{\sqrt{K}, 0, ..., 0\right\} \in \mathbb{H}^{d,K}$. Finally, using the exponential map, we map the vector $\mathbf{z}^0$ to the hyperbolic space, obtaining the initial hyperbolic embedding for users and items. The process is as follows:

$$\mathbf{z}_i^0 = (0, \mathbf{x}_i), \qquad \mathbf{z}_u^0 = (0, \mathbf{x}_u), \tag{8}$$

$$\mathbf{e}_i^0 = \exp_{\mathbf{o}}(\mathbf{z}_i^0), \qquad \mathbf{e}_u^0 = \exp_{\mathbf{o}}(\mathbf{z}_u^0), \tag{9}$$

Hyperbolic Neighbor Aggregation. We believe that the representation of each user consists of two parts: their historical interactions and the features of similar users. Similarly, the attractiveness of each item can also be divided into two parts: the users who interact with it and the items that are similar to it. Based on this, we propose an aggregator that combines historical behaviors and semantic neighborhood information to enhance the modeling of user and item embeddings. Since user and item modeling are symmetric, we focus on user modeling for illustration in the following discussion.

For the first aggregation, it accounts for the user's historical behaviors. Since the importance of interacted items varies for each user, we adopt an attention mechanism to extract the user's core interests. Specifically, we apply an attention mechanism based on hyperbolic distance followed by the exponential map to aggregate historical interaction information into a hyperbolic user representation, formulated as:

$$a_{u_m i_n} = \frac{\exp(-d_{\mathcal{L}}^K(\mathbf{e}_{u_m}, \mathbf{e}_{i_n})/\tau)}{\sum_{i_{n'} \in I_{u_m}} \exp(-d_{\mathcal{L}}^K(\mathbf{e}_{u_m}, \mathbf{e}_{i_{n'}})/\tau)}, \tag{10}$$

$$\mathbf{e}_{u_m}^I = \exp_{\mathbf{o}}^K\Big(\sum_{i_n \in I_{u_m}} a_{u_m i_n} \cdot \log_{\mathbf{o}}^K(\mathbf{e}_{i_n})\Big) \tag{11}$$

where I_{u_m} is the item set that user u_m interacts with. $\mathbf{e}_{u_m}^I$ and $\mathbf{e}_{i_n}$ are the hyperbolic representations of user u_m and item i_n, respectively. τ is the temperature parameter.

Through the attention mechanism, we can extract effective user representations. However, since there may be extensive conformity behavior in users' historical interactions, the learned user representations cannot accurately reflect their intrinsic interests. Therefore, we further adopt a debiased method in user representations learning , hoping to reduce the user's conformity bias as much as possible.

To better align the learned user representations with their true preferences, we plan to improve it from two aspects. On the one hand, we will implement Interest-Driven Weight Enhancement to increase the weight of items that users are really interested in. On the other hand, we will apply Conformity-Driven Weight Reduction to reduce the weight of items that are interacted with because of conformity mentality.

Interest-Driven Weight Enhancement aims to describe the degree to which each interacted item meets the user's true interest. Specifically, we evaluate the importance of each interacted item by calculating the similarity score between each interacted item and the entire contextual interaction history as Formula 12. The higher the item similarity score, the better the item reflects the user's true preferences, and vice versa.

$$\gamma_{u_m i_n} = \sum_{i_{n'} \in I_{u_m}} (\log_{\mathbf{o}}^K(\mathbf{e}_{i_n}))^T \cdot W \cdot \log_{\mathbf{o}}^K(\mathbf{e}_{i_{n'}}) \tag{12}$$

where $W \in \mathbb{R}^{d \times d}$ is a parameter matrix. The larger $\gamma_{u_m i_n}$ is, the more significant i_n is in the interaction history of user u_m.

Conformity-Driven Weight Reduction aims to reduce the weight of interacted items caused by conformity mentality. Although it is difficult to distinguish the real reason for user clicks, items with a large amount of interactions are more likely to be popular items, and the behavior of clicking on these popular items is more likely to be influenced by the conformity mentality. Therefore, we use the degree of the interacted items to simulate their popularity and try to reduce the weight of the items with greater interaction for target users.

$$\delta_{u_m i_n} = \frac{1}{(c_{i_n} + 1)^b} \tag{13}$$

where c_{i_n} denotes the degree of interacted item i_n. b is the power-law exponent, typically satisfying $b > 1$. In this way, items with fewer interactions are more important to the target user, so the corresponding $\delta_{u_m i_n}$ is larger.

Afterwards, we incorporate the Interest-Driven Weight $\gamma_{u_m i_n}$ and the Conformity-Driven Weight $\delta_{u_m i_n}$ on the basis of the original attention weight $a_{u_m i_n}$:

$$a_{u_m i_n}{}' = Soft\max\left(a_{u_m i_n} \cdot \gamma_{u_m i_n} \cdot \delta_{u_m i_n}\right) \tag{14}$$

Then, we re-weight the representation of all interacted items to generate a new user representation that reflects the user's intrinsic preferences more accurately:

$$\mathbf{e}_{u_m}^I = \exp_{\mathbf{o}}^K\Big(\sum_{i_n \in I_{u_m}} a_{u_m i_n}{}' \cdot \log_{\mathbf{o}}^K(\mathbf{e}_{i_n})\Big) \tag{15}$$

For the second aggregation, it accounts for the user's semantic neighbors. According to the semantic graph we constructed in the previous subsection, we aggregate the semantic neighbors for each user:

$$w_{u_m u_p}{}' = Soft\max\left(w_{u_m u_p}\right) \tag{16}$$

$$\mathbf{e}_{u_m}^U = \exp_{\mathbf{o}}^K\Big(\sum_{u_p \in U_{u_m}} w_{u_m u_p}{}' \cdot \log_{\mathbf{o}}^K(\mathbf{e}_{u_p})\Big) \tag{17}$$

where $w_{u_m u_p}$ denotes the weight of user u_p to user u_m. U_{u_m} represents the neighbor set of user u_m in the semantic graph.

Finally, we aggregate the self-information in the l-th layer $\mathbf{e}^l_{u_m}$, historical interaction representation $\mathbf{e}^I_{u_m}$ and semantic neighbor representation $\mathbf{e}^U_{u_m}$ as the user's representation in the $(l+1)$-th convolution layer.

$$\mathbf{e}^{l+1}_{u_m} = \exp^K_{\mathbf{o}}\left(\log^K_{\mathbf{o}}(\mathbf{e}^l_{u_m}) + \log^K_{\mathbf{o}}(\mathbf{e}^I_{u_m}) + \log^K_{\mathbf{o}}(\mathbf{e}^U_{u_m})\right) \quad (18)$$

Furthermore, to mitigate the problem of over-smoothing and ensure that the learned representations better capture the structural and semantic differences in the graph, we aggregate the representations of all L convolution layers as Formula 19. Due to resource limitations, we set L to 1 in this paper.

$$\mathbf{e}_{u_m} = \exp^K_{\mathbf{o}}\left(\sum_{l=0}^{L} \log^K_{\mathbf{o}}(\mathbf{e}^l_{u_m})\right) \quad (19)$$

Prediction and Optimization

Prediction. In this paper, we use the hyperbolic distance defined in Equation 4 to measure the similarity between the target user u_m and the candidate items i_c, and obtain the click prediction score y as follows:

$$y_{u_m i_c} = -d^K_{\mathcal{L}}(\mathbf{e}_{u_m}, \mathbf{e}_{i_c})^2, \quad (20)$$

Optimization. In DHRec, we adopt the adaptive margin loss proposed by HICF (M. Yang, Li, et al., 2022) to optimize our model.

$$\mathcal{L} = \sum_{m=1}^{M} \sum_{(u_m, i_c, i_{c'}) \in \mathcal{T}_m} \max(y_{u_m i_{c'}} - y_{u_m i_c} + m_{u_m i_c}, 0), \quad (21)$$

where $\mathcal{T}_m = \{(u_m, i_c, i_{c'}) | u_m \in \mathcal{U}, i_c, i_{c'} \in I, X_{u_m i_c} = 1, X_{u_m i_{c'}} = 0\}$. $m_{u_m i_c}$ denotes the adaptive margin which is learned by the positive pair (u_m, i_c):

$$m_{u_m i_c} = \sigma\left(\frac{d^K_{\mathcal{L}}(\mathbf{e}_{u_m}, \mathbf{o})^2 + d^K_{\mathcal{L}}(\mathbf{e}_{i_c}, \mathbf{o})^2 - d^K_{\mathcal{L}}(\mathbf{e}_{u_m}, \mathbf{e}_{i_c})^2}{\mathbf{e}_{u_m,0}\mathbf{e}_{i_c,0}}\right). \quad (22)$$

where the denominator is for normalization. $\mathbf{e}_{u_m,0} > 1$ and $\mathbf{e}_{i_c,0} > 1$ denote the zeroth coordinate element in e_{u_m} and e_{i_c} , respectively. σ is the *sigmoid* function.

Experiments

Experimental Settings

Datasets. In this paper, we use three general datasets to evaluate the effectiveness of DHRec: Amazon-CDs [1], Amazon-Books [2] and Yelp2020 [3]. The statistics are summarized in Table 1. These datasets vary in size and sparsity providing a robust performance measure. The training set, validation set, and test set are randomly split with the ratio of 70%/10%/20%. All models are evaluated on the test set using two widely used metrics Recall@K (R@K) and NDCG@K (N@K). In DHRec, we set $K \in \{10, 20\}$ to report the average values of the metrics for all users when testing.

[1] https://jmcauley.ucsd.edu/data/amazon/

[2] https://jmcauley.ucsd.edu/data/amazon/

[3] https://www.yelp.com/dataset

Table 1: Information on datasets.

Datasets	#Users	#Items	#Interactions	Density
Amazon-CDs	22947	18395	422301	0.100%
Amazon-Books	52406	41264	1861118	0.086%
Yelp2020	91174	45063	1940014	0.047%

Baselines. To evaluate the performance of our model towards the recommendation, we select nine SOTA baselines from three perspectives: the first five are based on Euclidean space, with the first two using MF (e.g., BPRMF (Rendle, Freudenthaler, Gantner, & Schmidt-Thieme, n.d.), WRMF (Hu, Koren, & Volinsky, 2008)) to uncover latent user-item relationships, and the next three utilizing GNNs (e.g., NGCF (X. Wang, He, Wang, Feng, & Chua, 2019), LightGCN (He et al., 2020a), DGCF (X. Wang et al., 2020)) to capture complex user-item interactions. The last four baselines are based on hyperbolic space (e.g., HGCF (Sun et al., 2021), LGCF (L. Wang et al., 2021), HRCF (M. Yang, Zhou, et al., 2022) and HICF (M. Yang, Li, et al., 2022), which better captures hierarchical relationships in recommendation tasks.

Experiment Settings. We set embedding dimension to 50, batch size to 10000, training epochs to 500. Riemannian SGD (Bonnabel, 2013) is used with the learning rate varies from $\{0.01, 0.001, 0.0001\}$. Then we set the hyper-parameters of the time parameter $t = 100$, temperature parameter $\tau = 0.1$, and turn the semantic neighbor size K in the ranges of $\{3, 5, 8, 10\}$. The best settings for the hyper-parameters in all baselines follow their original papers.

Results and Analysis

Performance Comparison. To demonstrate the effectiveness and superiority of our proposed model DHRec, we compare DHRec with all baselines on three real-world datasets, and the results are shown in Table 2. The best performance scores are bold, while the strongest baselines are highlighted with an underline. *%Improv.* denotes the relative improvement over the strongest baseline. From the results, we have the following observations:

Firstly, DHRec outperforms all baselines across the three datasets. The highest improvements reach 11.04% and 10.90% on Recall and NDCG metrics when compared with the best baseline HICF (M. Yang, Li, et al., 2022), respectively, highlighting its impressive effectiveness. Specifically, compared to the best baseline, DHRec achieves the highest performance improvement on Recall and NDCG by 4.26% and 7.55% on Amazon-CDs, by 10.67% and 11.04% on Amazon-Books, and by 9.07% and 10.90% on Yelp2020, which validates the superiority of our approach.

Secondly, the experimental results show that almost all GNNs-based models (NGCF, LightGCN and DGCF) outperform MF-based models (BPRMF and WRMF). This demonstrates GNNs' powerful representation capabilities in uncovering deeper structural and semantic information within the user-item interaction graph in CF recommendation tasks.

Table 2: Results on datasets.

Methods	Amazon-CDs				Amazon-Books				Yelp2020			
	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20
BPRMF	0.0666	0.1036	0.0553	0.0672	0.0666	0.0998	0.0537	0.0645	0.0325	0.0556	0.0283	0.0512
WRMF	0.0762	0.1147	0.0631	0.0756	0.0609	0.0927	0.0511	0.0614	0.047	0.0793	0.0372	0.0506
NGCF	0.071	0.1092	0.0586	0.071	0.0637	0.0974	0.0523	0.0632	0.0458	0.0764	0.0405	0.0513
LightGCN	0.0834	0.1259	0.0688	0.0825	0.0827	0.1215	0.069	0.0815	0.0522	0.0866	0.0461	0.0582
DGCF	0.0765	0.1138	0.0633	0.0754	0.0714	0.1069	0.0591	0.0706	0.0509	0.0786	0.0408	0.0529
HGCF	0.0962	0.1455	0.0751	0.0909	0.0867	0.1318	0.0869	0.1022	0.0543	0.0884	0.0458	0.0585
LGCF	0.0996	0.1503	0.078	0.0945	0.0899	0.136	0.0906	0.1063	0.0573	0.0946	0.0485	0.0612
HRCF	0.1003	0.1503	0.0785	0.0947	0.09	0.1364	0.0902	0.106	0.0537	0.0898	0.0468	0.0594
HICF	<u>0.1079</u>	<u>0.1586</u>	<u>0.0848</u>	<u>0.101</u>	<u>0.0965</u>	<u>0.1449</u>	<u>0.0978</u>	<u>0.1142</u>	<u>0.057</u>	<u>0.0948</u>	<u>0.0502</u>	<u>0.0633</u>
DHRec	**0.1125**	**0.1649**	**0.0912**	**0.1083**	**0.1068**	**0.1572**	**0.1086**	**0.1253**	**0.062**	**0.1034**	**0.0553**	**0.0702**
%Improv.	+4.26	+3.97	+7.55	+7.23	+10.67	+8.49	+11.04	+9.72	+8.77	+9.07	+10.16	+10.90

Table 3: Results on datasets.

Component				Amazon-CDs				Amazon-Books				Yelp2020			
IG	SG	DL	SL	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20
✓				0.0941	0.1387	0.0728	0.0896	0.0936	0.1416	0.0925	0.1002	0.0523	0.0845	0.0412	0.0475
✓		✓		0.0998	0.1468	0.0798	0.0976	0.0993	0.1457	0.0978	0.1096	0.0578	0.0897	0.0469	0.0532
✓	✓			0.1014	0.1473	0.0819	0.0997	0.0985	0.1423	0.0962	0.1069	0.0536	0.0905	0.0421	0.0574
✓	✓	✓		0.1087	0.1538	0.0875	0.1028	0.1021	0.1489	0.1017	0.1126	0.0583	0.0972	0.0489	0.0657
✓	✓	✓	✓	0.1125	0.1649	0.0912	0.1083	0.1068	0.1572	0.1086	0.1253	0.062	0.1034	0.0553	0.0702

Additionally, nearly all hyperbolic space-based models (HGCF, LGCF, HRCF and HICF) outperform Euclidean space-based models (BPRMF, WRMF, NGCF, LightGCN and DGCF). This indicates the stronger expressive power of hyperbolic space in modeling user and item representations, as it better captures hierarchical structures and relationships within the power-law distributed data, which are critical for CF recommendation tasks.

Finally, when comparing our model DHRec to the second strongest baseline HICF, we observe that DHRec shows a more significant improvement on the Amazon-Books datasets and the Yelp2020 dataset than on the Amazon-CDs dataset. We attribute this improvement to the construction of the user and item semantic graph in DHRec, which enables the model to better capture semantic neighbor information and enhances its ability to handle sparse datasets.

Ablation Study. To evaluate the effectiveness of each component, we performed an ablation study. The results are shown in Table 3. In this table, IG, SG, DL, and SL represent Interaction Graph, Semantic Graph, Debiased Learning in the interaction graph, and Self-Loop in Formula 18, respectively.

The experimental results reveal that removing any of these components results in a significant decline in recommendation performance across all three datasets. Specifically, as seen in the table, removing any component leads to lower recall and NDCG values, highlighting each component critical role in improving recommendation quality. Then, excluding the SG component or the DL component results in noticeable performance drops, indicating their significance in learning the importance of contextual semantic information and addressing representation distortions problems introduced by conformity bias. The SL component also contributes significantly, as evidenced by its inclusion improving performance in terms of both Recall and NDCG. From a human cognition perspective, we know that while neighbor information is important, it is the information from the self that is central. Therefore, self-loop is crucial in our recommendation model. These results demonstrate that all components are essential for achieving optimal recommendation performance, confirming the effectiveness of our model's design.

Conclusion

In this paper, we propose a novel Debiased Hyperbolic Recommendation model, DHRec, which addresses the challenges posed by the mismatch between data volume space and the embedding space in Euclidean space, as well as conformity bias in CF recommendation. By leveraging hyperbolic space, we effectively capture the complex hierarchical structures in user-item interactions, enhancing the quality of user and item representations. Additionally, by incorporating semantic graphs and interaction graph, we aggregate the contextual information and interaction history. Moreover, by adjusting the weights of neighbor nodes, we significantly mitigate the conformity bias, leading to more accurate recommendations. Extensive experiments demonstrate the effectiveness and superiority of DHRec on three real-world datasets.

References

Bonnabel, S. (2013, September). Stochastic gradient descent on riemannian manifolds. *IEEE Transactions on Automatic Control*, *58*(9), 2217–2229. Retrieved from http://dx.doi.org/10.1109/TAC.2013.2254619 doi: 10.1109/tac.2013.2254619

Chen, J., Dong, H., Wang, X., Feng, F., Wang, M., & He, X. (2023, February). Bias and debias in recommender system: A survey and future directions. *ACM Trans. Inf. Syst.*, *41*(3). Retrieved from https://doi.org/10.1145/3564284 doi: 10.1145/3564284

Cheng, H.-T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., … Shah, H. (2016, June). *Wide & deep learning for recommender systems* (No. arXiv:1606.07792). arXiv. doi: 10.48550/arXiv.1606.07792

Gulcehre, C., Denil, M., Malinowski, M., Razavi, A., Pascanu, R., Hermann, K. M., … de Freitas, N. (2018, May). *Hyperbolic attention networks* (No. arXiv:1805.09786). arXiv. doi: 10.48550/arXiv.1805.09786

He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., & Wang, M. (2020a). Lightgcn: Simplifying and powering graph convolution network for recommendation. *CoRR*, *abs/2002.02126*. Retrieved from https://arxiv.org/abs/2002.02126

He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., & Wang, M. (2020b, July). 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* (pp. 639–648). Virtual Event China: ACM. doi: 10.1145/3397271.3401063

Hu, Y., Koren, Y., & Volinsky, C. (2008). Collaborative filtering for implicit feedback datasets. In *2008 eighth ieee international conference on data mining* (p. 263-272). doi: 10.1109/ICDM.2008.22

Koren, Y., Bell, R., & Volinsky, C. (2009, August). Matrix Factorization Techniques for Recommender Systems. *Computer*, *42*(8), 30–37. doi: 10.1109/MC.2009.263

Li, A., Yang, B., Huo, H., Chen, H., Xu, G., & Wang, Z. (2023, September). Hyperbolic Neural Collaborative Recommender. *IEEE Transactions on Knowledge and Data Engineering*, *35*(9), 9114–9127. doi: 10.1109/TKDE.2022.3221386

Nickel, M., & Kiela, D. (2018). Learning continuous hierarchies in the lorentz model of hyperbolic geometry. *CoRR*, *abs/1806.03417*. Retrieved from http://arxiv.org/abs/1806.03417

Ravasz, E., & Barabási, A.-L. (2003, February). Hierarchical organization in complex networks. *Physical Review E*, *67*(2). Retrieved from http://dx.doi.org/10.1103/PhysRevE.67.026112 doi: 10.1103/physreve.67.026112

Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (n.d.). BPR: Bayesian personalized ranking from implicit feedback.

Sun, J., Cheng, Z., Zuberi, S., Perez, F., & Volkovs, M. (2021, June). HGCF: Hyperbolic Graph Convolution Networks for Collaborative Filtering. In *Proceedings of the Web Conference 2021* (pp. 593–601). New York, NY, USA: Association for Computing Machinery. doi: 10.1145/3442381.3450101

Tu, L., Meng, S., Qi, L., Xu, X., & Zhang, X. (2024, July). Hyperbolic contrastive learning with second order sampling for collaborative filtering. In *2024 IEEE International Conference on Web Services (ICWS)* (pp. 281–290). doi: 10.1109/ICWS62655.2024.00050

Wang, L., Hu, F., Wu, S., & Wang, L. (2021, October). Fully hyperbolic graph convolution network for recommendation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management* (pp. 3483–3487). Virtual Event Queensland Australia: ACM. doi: 10.1145/3459637.3482109

Wang, S., Guo, S., Wang, L., Liu, T., & Xu, H. (2023, July). HDNR: A hyperbolic-based debiased approach for personalized news recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 259–268). Taipei Taiwan: ACM. doi: 10.1145/3539618.3591693

Wang, X., He, X., Wang, M., Feng, F., & Chua, T. (2019). Neural graph collaborative filtering. *CoRR*, *abs/1905.08108*. Retrieved from http://arxiv.org/abs/1905.08108

Wang, X., Jin, H., Zhang, A., He, X., Xu, T., & Chua, T. (2020). Disentangled graph collaborative filtering. *CoRR*, *abs/2007.01764*. Retrieved from https://arxiv.org/abs/2007.01764

Yang, M., Li, Z., Zhou, M., Liu, J., & King, I. (2022, August). HICF: Hyperbolic Informative Collaborative Filtering. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 2212–2221). Washington DC USA: ACM. doi: 10.1145/3534678.3539475

Yang, M., Zhou, M., Liu, J., Lian, D., & King, I. (2022, May). *HRCF: Enhancing Collaborative Filtering via Hyperbolic Geometric Regularization* (No. arXiv:2204.08176). arXiv. doi: 10.48550/arXiv.2204.08176

Yang, Y., Wu, L., Zhang, K., Hong, R., Zhou, H., Zhang, Z., … Wang, M. (2024, December). Hyperbolic Graph Learning for Social Recommendation. *IEEE Transactions on Knowledge and Data Engineering*, *36*(12), 8488–8501. doi: 10.1109/TKDE.2023.3343402

Zhang, L., & Wu, N. (2024, October). 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* (pp. 3186–3196). New York, NY, USA: Association for Computing Machinery. doi: 10.1145/3627673.3679701