HAKG: Hierarchy-Aware Knowledge Gated Network for Recommendation

Yuntao DU

Xinjun ZHU

Lu CHEN

Baihua ZHENG
Singapore Management University, bhzheng@smu.edu.sg

Yunjun GAO

Citation

HAKG: Hierarchy-Aware Knowledge Gated Network for Recommendation

Yuntao Du†, Xinjun Zhu§, Lu Chen†, Baihua Zheng♯, and Yunjun Gao†

†College of Computer Science, Zhejiang University, Hangzhou, China

§School of Software, Zhejiang University, Ningbo, China

♯School of Computing and Information Systems, Singapore Management University, Singapore

{1ytdu, 2xjzhu, 3luchen, 5gaoyj}@zju.edu.cn   4bhzheng@smu.edu.sg

ABSTRACT

Knowledge graph (KG) plays an increasingly important role to improve the recommendation performance and interpretability. A recent technical trend is to design end-to-end models based on information propagation mechanism. However, existing propagation-based methods fail to (1) model the underlying hierarchical structures and relations, and (2) capture the high-order collaborative signals of items for learning high-quality user and item representations.

In this paper, we propose a new model, called *Hierarchy-Aware Knowledge Gated Network* (HAKG), to tackle the aforementioned problems. Technically, we model users and items (that are captured by a user-item graph), as well as entities and relations (that are captured in a KG) in hyperbolic space, and design a hyperbolic aggregation scheme to gather relational contexts over KG. Meanwhile, we introduce a novel angle constraint to preserve characteristics of items in the embedding space. Furthermore, we propose a novel dual item embeddings design to represent and propagate collaborative signals and knowledge associations separately, and leverage the gated aggregation to distill discriminative information for better capturing user behavior patterns. Experimental results on three benchmark datasets show that, HAKG achieves significant improvement over the state-of-the-art methods like CKAN, Hyper-Know, and KGIN. Further analyses on the learned hyperbolic embeddings confirm that HAKG offers meaningful insights into the hierarchies of data.

CCS CONCEPTS

• **Information systems** → **Recommender systems**.

KEYWORDS

Recommendation, Graph Neural Network, Knowledge Graph

ACM Reference Format:
Yuntao Du†, Xinjun Zhu§, Lu Chen†, Baihua Zheng♯, and Yunjun Gao†. 2022. HAKG: Hierarchy-Aware Knowledge Gated Network for Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22), July 11–15, 2022, Madrid,*

(a) Knowledge-aware recommendation   (b) Distribution of interactions
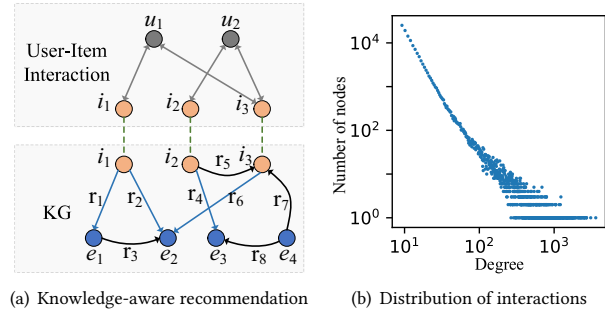
Figure 1: (a) An example of knowledge-aware recommendation. The blue lines indicate the hierarchical relations, while the black lines denote the non-hierarchical relations. (b) Degree distribution of Alibaba-iFashion dataset.

Spain. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/XXXXXX. XXXXXX

1 INTRODUCTION

In the era of information explosion, recommender systems have become an essential part of Internet applications to provide personalized information services. Traditional recommender systems that are based on collaborative filtering [18, 24, 43] usually suffer from data sparsity and cold-start problems. Recently, knowledge graph (KG), which provides various real-world facts related to items via relations, has demonstrated an impressive ability to alleviate cold-start issue and improve the explainability of recommendations.

Learning user and item representations from auxiliary KG has become the term of knowledge-aware recommendation. Early studies [1, 39, 49] directly integrate knowledge graph embeddings with items to enhance their representations. Some subsequent studies [5, 17, 21, 45] enrich the interactions via meta-paths from users to items for better identifying user-item connectivities. However, in order to obtain informative paths, these methods suffer from labor-intensive process [45], poor generalization [17, 52], and unstable performance [48]. Recently, the success of Graph Neural Networks (GNNs) [14, 23, 35] has inspired the community to develop end-to-end models based on the information aggregation schemes [9, 34, 38, 40, 42, 44, 46]. The key idea is to iteratively propagate high-order information over KG, which could effectively integrate multi-hop neighbors into representations and hence improve the recommendation performance.

Although existing propagation-based methods are able to achieve good performance, we would like to highlight that they all fail to model the following two important factors.

- **Hierarchical Structures and Relations**. Existing methods model both user-item interactions and KG in Euclidean space, while both data structures exhibit a highly non-Euclidean latent anatomy. Specifically, the user-item interactions typically follow the power-law distribution (as shown in Figure 1(b)), indicating the underlying hierarchical structures [9, 36, 50]. Meanwhile, hierarchical information is ubiquitous in real-world KGs, since human knowledge is organized hierarchically [2, 6]. Euclidean-based methods are insufficient to capture the intrinsic hierarchical structures of the data, since they suffer from a high distortion when embedding hierarchical data [7, 12, 32]. Moreover, none of those methods considers KG relations at a finer-grained level of hierarchies. They all conveniently ignore an important fact, *i.e.,* hierarchical relations (blue lines in Figure 1(a)) and non-hierarchical relations (black lines in Figure 1(a)) are **not** of equal importance for characterizing items. Taking the outfit recommendation in Figure 1(a) as an example. Hierarchical relation r_1 offers complementary information to the cloth i_1 in the aspect that i_1 is the jeans (e_1) while non-hierarchical relation r_5 means that cloth i_2 and cloth i_3 can compose an outfit. Thus, hierarchical relations can indicate the attributes of items while non-hierarchical relations only reveal the relatedness between entities. Ignoring the hierarchical structures and relations limits model's expressive power.
- **High-order Collaborative Signals of Items.** In existing studies, the item aggregation schemes are mostly KG-oriented, that is, recursively collecting the knowledge associations from KG without considering the collaborative signals from users [44, 46], or blindly mixing the heterogeneous information from neighboring nodes (users or entities) in the Unified Knowledge Graph (UKG) [34, 42]. They fail to preserve the crucial high-order collaborative signals of items (*e.g.,* co-occurrence relationship), which are latent in user-item interactions and play an important role in learning user preference from behavior aspect. Take Figure 1(a) as an example. The path $i_1 \rightarrow u_1 \rightarrow i_3 \rightarrow u_2 \rightarrow i_2$ indicates the long-range connectivity between item i_1 and item i_2: since users u_1 and u_2 share the same interest (*i.e.,* they both like item i_3), the items i_1 and i_2 which are favored by user u_1 and user u_2 respectively may be similar to some extent. Therefore, such KG-oriented aggregation schemes are insufficient to capture the high-order collaborative signals for comprehensive item representations.

To tackle the aforementioned challenges, a new latent space with a smaller embedding distortion is required. Hyperbolic geometry offers an ideal alternative, as it not only enables embeddings with much smaller distortion [7, 12] but also naturally preserves the data hierarchy [6, 50]. The key property of hyperbolic space is that, unlike Euclidean space, it expands exponentially rather than polynomially. This allows hyperbolic space to have much larger data capacity than Euclidean space, and hence, it has "sufficient" room to better preserve the distance between hierarchical data, as shown in the left of Figure 2. Besides, hyperbolic space can be viewed as a continuous version of trees, which makes it perfect to model hierarchical tree-like data due to their analogous structure. Thus, we propose Hierarchy-Aware Knowledge Gated Network (HAKG), a new hyperbolic knowledge-aware recommendation model with two key components, which can effectively capture and model the
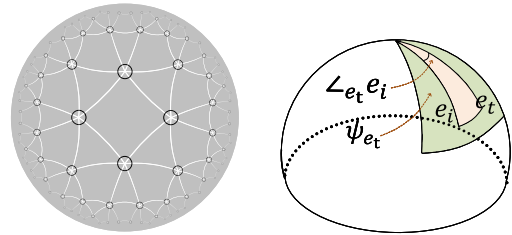

Figure 2: (1) Hyperbolic space expands exponentially. (2) Illustration of Hyperbolic cones.

aforementioned two important factors that are *not* modeled by any existing methods.

- **Hierarchy-Aware Modeling.** To better capture the underlying hierarchical structures, we map user and item embeddings as well as entity and relation embeddings to hyperbolic space. We also design a new hyperbolic relation-transitive aggregation mechanism to capture the relation dependencies carried by neighbors in hyperbolic space. Moreover, a novel angle constraint is introduced for hierarchical relations, which is able to better preserve items' attributes information in the embedding space and thus improve the representational capacity and expressiveness.
- **Gated Aggregation with Dual Embeddings.** Unlike previous KG-oriented aggregation strategies, we view user-item interactions and KG as two different information channels, and develop different aggregation strategies for the two channels. Besides, since items can serve as a natural bridge to connect the two information channels, we use *dual* embedding instead of *single* embedding for items to represent and propagate each channel's information separately, so as to better capture the holistic semantics of items. Moreover, an information gated mechanism is introduced to adaptively fuse the two types of semantics for better identifying user behavior patterns.

To this end, the newly proposed HAKG framework is designed to i) leverage the expressiveness of hyperbolic geometry for better hierarchical modeling, and ii) effectively capture the holistic semantics of items. We conduct extensive experiments on three real-world datasets to evaluate the performance of HAKG and existing methods. Experimental results show that our HAKG significantly outperforms all the start-of-the-art methods such as CKAN [46], Hyper-Know [29], and KGIN [44]. Furthermore, HAKG is able to offer both behavior and attribute similarity scores of unseen user-item interactions, and reveal the underlying hierarchical structures and relations of data in the embedding space for better model expressiveness. In summary, our contributions are as follows:

- We present knowledge-aware recommendation from a new perspective by taking the hierarchy and high-order items' collaborative signals into consideration.
- We embed users and items, as well as entities and relations in hyperbolic space, and design a new hyperbolic relation-transitive aggregation in KG to preserve the relation dependencies among neighbors. Besides, an angle constraint is introduced for hierarchical relations to profile items with the attributes information in the embedding space.
- We incorporate the dual item embeddings design to i) better represent and propagate collaborative signals and knowledge

associations simultaneously, and ii) more effectively leverage the information gate mechanism to control discriminative signals towards the users' preference, so as to offer similarity scores of each interaction from both behavior and attribute aspects.

- We conduct extensive experiments on three public benchmark datasets to demonstrate the superiority of HAKG.

2 PROBLEM FORMULATION

We first introduce the data structures related to our studied problem, and then formulate our task.

User-Item Bipartite Graph. In this paper, we focus on learning the user preference from implicit feedback [33]. To be more specific, the behavior data (*e.g.,* click and review) involves a set of users $\mathcal{U} = \{u\}$ and a set of items $\mathcal{I} = \{i\}$. We model user-item interactions as a bipartite graph $\mathcal{G}_b = \{(u, i)|u \in \mathcal{U}, i \in \mathcal{I}\}$ where each (u, i) pair indicates that user u has interacted with item i.

Knowledge Graph (KG). KGs are collections of real-world facts, such as item attributes, concepts, or external commonsense. Let $\mathcal{T}$ be the triplet set, $\mathcal{E}$ be a set of entities, and $\mathcal{R}$ be the relation set, which involves relations in both canonical and inverse directions (*e.g., compose* and *composed-of*). Let KG be a heterogeneous graph $\mathcal{G}_k = \{(h, r, t)|h, t \in \mathcal{E}, r \in \mathcal{R}\}$, where each triplet $(h, r, t) \in \mathcal{T}$ means that there is a relation r between head entity h and tail entity t. For example, a triplet (*jeans, brand, Levi's*) indicates that the *brand* of the *jeans* is *Levi's*. As we assume all the items appear in KG as entities (i.e., $\mathcal{I} \subset \mathcal{E}$), a common assumption made by all existing knowledge-aware recommendation systems [41, 42, 44], we can link items in user-item graph with entities in KG to offer auxiliary semantics to interactions.

Task Description. Given a user-item bipartite graph $\mathcal{G}_b$ and a KG $\mathcal{G}_k$, our task of knowledge-aware recommendation is to predict how likely that a user would adopt an item that she has never engaged with.

3 METHODOLOGY

In this section, we first introduce the basic concepts about hyperbolic space, and then detail two key components of HAKG, *i.e., hierarchy-aware modeling* and *gated aggregation with dual embeddings*. The former embeds users and items as well as entities and relations in hyperbolic space. It adopts a novel hyperbolic relation-aware aggregation scheme over KG, and also introduces an angle constraint for learning attribute semantics of item in the embedding space. The latter leverages dual item embeddings to separately encode the high-order collaborative signals from user-item graph and the knowledge associations from KG, and distills useful semantics via the information gate mechanism for learning high-quality user representations.

3.1 Preliminaries

Hyperbolic geometry [3] is a non-Euclidean geometry with constant negative curvature. Compared with the Euclidean spaces, the amount of space covered by the hyperbolic geometry increases exponentially rather than polynomially *w.r.t.* the radius. This property allows us to effectively capture underlying hierarchical structures of user-item interactions and KG in hyperbolic space. In this work,

we use the Poincaré ball model with constant curvature $-c = -1$ for learning hyperbolic embeddings due to its feasibility for gradient optimization [32, 50]. In the following, we first introduce the necessary mathematical basis for the Poincaré ball model.

Poincaré Ball & Tangent Space. The definition domain of the Poincaré ball model is:

$$\mathbb{B} = \{(x_1, \ldots, x_n) : x_1^2 + \cdots + x_n^2 < \frac{1}{c}\} \tag{1}$$

in $\mathbb{R}^n$. The tangent space $\mathcal{T}_z\mathbb{B}$ at point z on $\mathbb{B}$ is a n-dimensional Euclidean space that best approximates $\mathbb{B}$ around z, which is useful to perform aggregation operations in hyperbolic space [7].

Exponential Map & Logarithmic Map. Hyperbolic space and tangent space can be bridged by exponential and logarithmic mappings. Specifically, the exponential map can map the tangent space $\mathcal{T}_z\mathbb{B}$ to the hyperbolic space $\mathbb{B}$, and the logarithmic map maps $\mathbb{B}$ to $\mathcal{T}_z\mathbb{B}$ conversely. In particular, the closed-form expressions for the two maps at point z are defined below.

$$\exp_z(\mathbf{x}) = \mathbf{z} \oplus \tanh(\frac{\|\mathbf{x}\|}{1 - \|\mathbf{z}\|^2})\frac{\mathbf{x}}{\|\mathbf{x}\|} \tag{2}$$

$$\log_z(\mathbf{y}) = (1 - \|\mathbf{z}\|^2) \cdot \tanh^{-1}(\| - \mathbf{z} \oplus \mathbf{y}\|)\frac{-\mathbf{z} \oplus \mathbf{y}}{\| - \mathbf{z} \oplus \mathbf{y}\|} \tag{3}$$

where $\oplus$ represents Möbius addition [12]:

$$\mathbf{x} \oplus \mathbf{y} = \frac{(1 + 2\langle\mathbf{x}, \mathbf{y}\rangle + \|\mathbf{y}\|^2)\mathbf{x} + (1 - \|\mathbf{x}\|^2)\mathbf{y}}{1 + 2\langle\mathbf{x}, \mathbf{y}\rangle + \|\mathbf{x}\|^2\|\mathbf{y}\|^2} \tag{4}$$

3.2 Hierarchy-Aware Modeling

Unlike the previous propagation-based studies [34, 40, 42, 44] that simply embed user-item graph and KG in Euclidean space, we aim to capture the non-Euclidean latent anatomy of data. Taking the Alibaba-iFishion dataset shown in Figure 1(b) as an example, the degree distribution of user-item interactions reveals an underlying hierarchical tree-like structure with power-law distribution. Meanwhile, KG also exhibits hierarchical patterns [2, 6]. However, the hierarchical property of data cannot be properly captured in Euclidean space because of the high-distortion [12, 32]. Thus, we embed users and items as well as entities and relations in hyperbolic space, where the hierarchical structures can be naturally preserved. We also design a new aggregation scheme in KG since previous Euclidean-based methods are not feasible in hyperbolic space.

Meanwhile, we argue that previous methods are unable to characterize items properly because they do not differentiate hierarchical relations from the rest, and *only* model KG relations at a coarse granularity. Specifically, hierarchical relations in KG often carry entities' attributes information (*e.g.,* brand and *fabric*), while non-hierarchical relations can only indicate the relatedness between entities (*e.g., match* and *similar to*). This motivates us to introduce an additional constraint for hierarchical triplets in the embedding space to improve the representational capacity and expressiveness.

3.2.1 ***Hyperbolic Relation-Transitive Aggregation.*** We first consider the aggregation scheme of KG in hyperbolic space. As an item i can be involved in multiple KG triplets, its neighborhood reflects the relational similarity between i and its connected entities to a certain degree. Formally, given a target item i in KG, aggregating local information from i's neighbors in KG can reveal related knowledge associations for i. Thus, we use $\mathcal{N}_i = \{(r, t)|(i, r, t) \in \mathcal{G}_k\}$ to
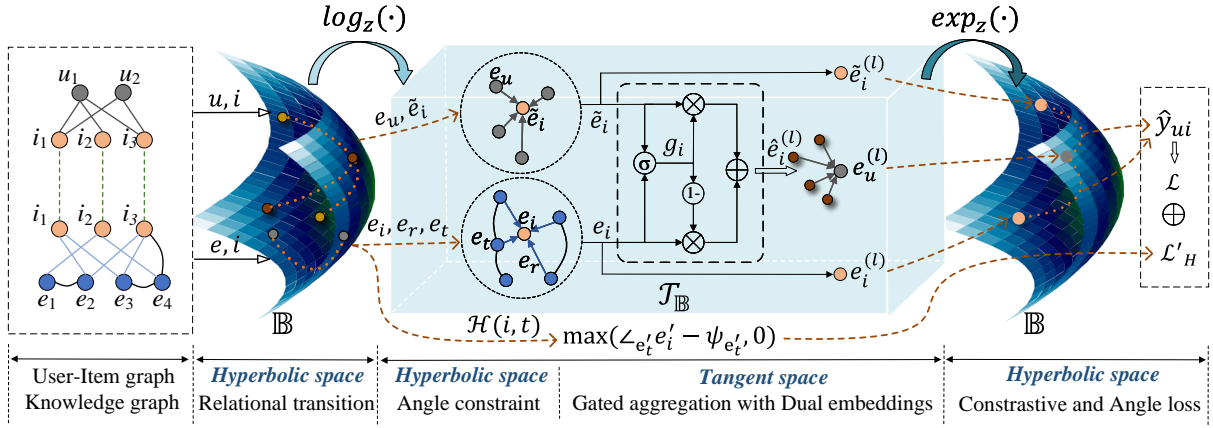
Figure 3: Illustration of the proposed HAKG framework.

represent the neighborhood entities and the first-order relations of item i in KG, and propose to integrate the relational context from neighborhood entities to generate the *knowledge representation* of item i:

$$\mathbf{e}_i^{(1)} = f_{\mathbb{B}}\Big(\{(\mathbf{e}_i^{(0)}, \mathbf{e}_r, \mathbf{e}_t^{(0)}) | (r,t) \in \mathcal{N}_i\}\Big) \qquad (5)$$

where $\mathbf{e}_i^{(1)} \in \mathbb{B}^d$ is the hyperbolic knowledge representation that collects the contextual information of item i from the first-order connectivity, and $f_{\mathbb{B}}(\cdot)$ is the hyperbolic aggregation function to extract and integrate information related to item i from each connection (i, r, e) in hyperbolic space. Previous studies [44] have shown that node-based aggregation cannot differentiate relational paths, which fails to preserve the relation dependencies carried by paths. Consequently, it is necessary to integrate relational paths into representations. Besides, Euclidean propagation-based methods mainly use mean aggregation to average the neighbor information. An analog of mean aggregation in hyperbolic space is the Fréchet mean [11], which, however, has no closed form solution. Instead, we model relational context in the mean aggregator as:

$$\mathbf{e}_i^{(1)} = \exp_{\mathbf{0}}\Big(\frac{1}{|\mathcal{N}_i|} \sum_{(r,t) \in \mathcal{N}_i} \log_{\mathbf{e}_i^{(0)}}(\mathbf{e}_t^{(0)} \oplus \mathbf{e}_r)\Big) \qquad (6)$$

where $\oplus$ is the Möbius addition operation, $\log(\cdot)$ and $\exp(\cdot)$ are the aforementioned exponential map and logarithmic map respectively, and $\mathbf{e}_t^{(0)}$ is the ID embedding of entity t in hyperbolic space. For each triplet (i, r, t) in KG, we devise a relational transition $\mathbf{e}_t^{(0)} \oplus \mathbf{e}_r$ by modeling the relation r as the vector translation of $\mathbf{e}_t^{(0)}$. To avoid the complex mean operation in hyperbolic space, we first map each relational context to the tangent space, as this is where the Euclidean approximation performs the best (*cf.* Figure 3); we then leverage the exponential map to map it back to hyperbolic space to obtain the representation $\mathbf{e}_i^{(1)}$. As a result, the aggregator is able to integrate relational message into target representations in hyperbolic space, and avoid intricate hyperbolic mean pooling via map operations. Analogously, we can obtain the representation $\mathbf{e}_e^{(1)}$ of each KG entity $e \in \mathcal{E}$.

We further stack more aggregation layers to explore the high-order knowledge associations for items. Technically, we recursively formulate the knowledge representations of item i after l layers as:

$$\mathbf{e}_i^{(l)} = \exp_{\mathbf{0}}\Big(\frac{1}{|\mathcal{N}_i|} \sum_{(r,t) \in \mathcal{N}_i} \log_{\mathbf{e}_i^{(l-1)}}(\mathbf{e}_t^{(l-1)} \oplus \mathbf{e}_r)\Big) \qquad (7)$$

3.2.2 Angle Constraint of Hierarchical triplets. Next, we focus on modeling the attributes information of items with hierarchical relations in KG. To ease our discussion, we assume that all the relations in KG are clustered into two types (*i.e.,* hierarchical and non-hierarchical), and the types of relations are given *a priori*. In fact, the types of relations are explicitly available in most KGs such as the three datasets used in our experiments. When such information is not available, there are two alternative options. One is to infer the "hierarchicalness" of a relation by widely used Krackhardt scores criteria [2, 25]; the other is to simply assume that item-connected entities are acted as the attributes of items, and thus, the relations are hierarchical. Our experiments to be reported in Section 4.3.2 show that the difference between the two options is negligible.

As mentioned earlier, hierarchical relations can reveal the attributes of items. However, above aggregation scheme cannot preserve this semantic in the embedding space since it simply aggregates all neighboring information into representations. To explicitly model items' characteristic information, we consider the hyperbolic entailment cone [13], which is a family of convex cones that can model the hierarchies with nested angular cones. Technically, we use $C_{\mathbf{x}}$ to denote the cone at apex $\mathbf{x}$, and define the cone width function (*i.e.,* half aperture[1] of cone $C_{\mathbf{x}}$) as $\psi_{\mathbf{x}} = \arcsin(K\frac{1 - \|\mathbf{x}\|^2}{\|\mathbf{x}\|})$, where $K \in \mathbb{R}$ is a hyperparameter (we follow [2] and set $K = 0.1$). Also, for $\mathbf{x}, \mathbf{y} \in \mathbb{B}^d$, we define the angle of $\mathbf{y}$ at $\mathbf{x}$ to be the angle between the half-lines $\overrightarrow{\mathbf{ox}}$ and $\overrightarrow{\mathbf{xy}}$, and denote it as $\angle_{\mathbf{x}}\mathbf{y}$:

$$\angle_{\mathbf{x}}\mathbf{y} = \arccos\Big(\frac{\langle \mathbf{x}, \mathbf{y}\rangle(1 + \|\mathbf{x}\|^2) - \|\mathbf{x}\|^2(1 + \|\mathbf{y}\|^2)}{\|\mathbf{x}\|\|\mathbf{x} - \mathbf{y}\|\sqrt{1 + \|\mathbf{x}\|^2\|\mathbf{y}\|^2 - 2\langle \mathbf{x}, \mathbf{y}\rangle}}\Big) \qquad (8)$$

The transitivity property of angular cones [13] guarantees that if $\mathbf{y}$ is in $C_{\mathbf{x}}$, $C_{\mathbf{y}}$ is also in $C_{\mathbf{x}}$. In other words, hyperbolic entailment cone forms a nested structure in embedding space, and the width (*aka.* angle) of the cone can naturally indicate the *"attribute"* semantics of embeddings, as shown in the right of Figure 2. To leverage the expressive power of hyperbolic entailment cone geometry, we first use $\mathcal{H} = \{(i, t) | (i, r, t) \in \mathcal{G}_k \wedge r \text{ is hierarchical}\}$ to represent the set of hierarchical entities in terms of hierarchical relations r in KG, and use Eq. (9) to define the angle constraint in embedding space:

$$\mathcal{L}_{\mathrm{H}} = \sum_{(i,t) \in \mathcal{H}} \max(\angle_{\mathbf{e}_t}\mathbf{e}_i - \psi_{\mathbf{e}_t}, 0) \qquad (9)$$

where $\mathbf{e}_t$ and $\mathbf{e}_i$ are the final knowledge representations of entity i and entity t respectively, defined in Eq. (15). The hinge loss on

[1]https://en.wikipedia.org/wiki/Cone

angle constraint suggests the angle of $\angle_{\mathbf{e}_t} \mathbf{e}_i$ to be smaller than the width of entailment cone $C_{\mathbf{t}}$, meaning that the embedding of item $\mathbf{e}_i$ should be "contained" inside of the embedding of entity $\mathbf{e}_t$ in the entailment cone geometry.

While such a constraint can ensure the attributes semantic with geometric angles, it also limits the representational capacity. For example, let item i be a jeans with multiple hierarchical relations (*e.g., brand* and *fabric*). As each of i's connected entity defines an entailment cone in hyperbolic space, the embedding of i can only be in their intersection area. Since items often have dozens of relations in KG, this intersection area could be extremely small, leading to possible collapse in the embedding space. Thus, we relax it by randomly assigning a corresponding subspace for the embedding of item i that satisfies the following constraint:

$$\mathcal{L}'_{\text{H}} = \sum_{(i,t)\in\mathcal{H}} \max(\angle_{\mathbf{e}'_t} \mathbf{e}'_i - \psi_{\mathbf{e}'_t}, 0) \tag{10}$$

where $\mathbf{e}'_i$ and $\mathbf{e}'_t$ are the subspace embeddings of $\mathbf{e}_i$ and $\mathbf{e}_t$ respectively by randomly masking some dimensions. In our current implementation, each dimension has a probability of 0.5 to be masked off, while this probability can be easily adjusted to cater for the application needs. By doing so, we only enforce the angle constraint in a subset of d hyperbolic planes, leaving room for expressing relatedness and other knowledge semantics of items.

3.3 Gated Aggregation with Dual embeddings

The second key component of HAKG framework is designed mainly to capture the high-order collaborative signals of items. Existing methods utilize KG-oriented aggregation strategies to collect the information related to an item from its neighboring nodes. Unfortunately, this approach either mixes the collaborative signals with knowledge associations such that they become indistinguishable [9, 42], or completely neglects the collaborative signals of items [44, 46] that results in suboptimal item representations. Thus, we propose to separately aggregate the two types of information with dual item embeddings to solve the limitation: (1) Knowledge item embeddings, which are used to characterize items with knowledge information from KG, as demonstrated in Section 3.2. (2) Collaborative item embeddings, which encode items' high-order collaborative signals such as co-occurrence relationship in user-item graph. Neighbored with the holistic representations of items, we further develop an information gate mechanism to adaptively distill useful information for users' aggregation. We illustrate our approach in Figure 3.

*3.3.1 **Collaborative Aggregation for Items.*** To explicitly encode the collaborative signals of items, we initialize new item representations $\tilde{\mathbf{e}}^{(0)}$ in hyperbolic space to items' collaborative information, which are latent in user-item interactions. We also use the neighborhood aggregation scheme to integrate an item's multi-hop neighbors into its representation for capturing the high-order connectivities. Given an item i in the user-item graph, we use $\tilde{\mathcal{N}}_i = \{u|(u,i)\in\mathcal{G}_b\}$ to represent all the users who have interacted with the item. To generate the l-th layer representation of item i, we recursively integrate the collaborative information from item

i's neighbor users:

$$\tilde{\mathbf{e}}_i^{(l)} = \exp_{\mathbf{0}} \left(\frac{1}{|\tilde{\mathcal{N}}_i|} \sum_{(u)\in\tilde{\mathcal{N}}_i} \log_{\mathbf{0}}(\mathbf{e}_u^{(l-1)}) \right) \tag{11}$$

where $\tilde{\mathbf{e}}_i^{(l)}$ is the collaborative embedding of item i after l layers, $\log(\cdot)$ and $\exp(\cdot)$ are the exponential map and logarithmic map that are used to map between Euclidean space and hyperbolic space, and $\mathbf{e}_u^{(l-1)}$ is the hyperbolic embedding of user u after $l-1$ layers in hyperbolic space, which we will detail below.

*3.3.2 **Information Gated Aggregation for Users.*** To distill useful information from neighbor items, we design a gated module which is able to generate discriminative signals for users' aggregation. Specifically, inspired by the design of GRU [10] that learns gating signals to control the update of hidden states, we propose to learn a fusion gate to adaptively control the combination of two different types of semantic item representations. For an item i, given its knowledge representation $\mathbf{e}_i^{(l)}$ and collaborative representation $\tilde{\mathbf{e}}_i^{(l)}$ in the l-th layer, we balance the contributions of the two different types of information with a learnable gating fusion unit:

$$\mathbf{g}_i^{(l)} = \sigma(W_1 \log_{\mathbf{0}}(\mathbf{e}_i^{(l)}) + W_2 \log_{\mathbf{0}}(\tilde{\mathbf{e}}_i^{(l)})) \tag{12}$$

$$\hat{\mathbf{e}}_i^{(l)} = \exp_{\mathbf{0}} \left(\mathbf{g}_i^{(l)} \cdot \mathbf{e}_i^{(l)} + (1 - \mathbf{g}_i^{(l)}) \cdot \tilde{\mathbf{e}}_i^{(l)} \right) \tag{13}$$

where $W_1, W_2 \in \mathbb{R}^{d\times d}$ are learnable transformation parameters, and $\sigma(\cdot)$ is the Sigmoid function. Notation $\mathbf{g}_i^{(l)} \in \mathbb{R}^d$ denotes the learned gate signals to balance the contributions of collaborative signals and knowledge association of item i, as shown in Figure 3. A high value of $\mathbf{g}_i^{(l)}$ indicates that the users interact with item i mainly because of the attributes of item i (*e.g.*, i is the jeans), rather than the behavior similarities among items (*e.g.*, the jeans i liked by user u is collaboratively relevant to other bottoms that user u likes). Let $\mathcal{N}_u = \{i|(u,i)\in\mathcal{G}_b\}$ denote the set of items that user u has interacted with. Then, we can formulate the representation of user u with distilled information:

$$\mathbf{e}_u^{(l)} = \exp_{\mathbf{0}} \left(\frac{1}{|\mathcal{N}_u|} \sum_{(i)\in\mathcal{N}_u} \log_{\mathbf{0}}(\hat{\mathbf{e}}_i^{(l-1)}) \right) \tag{14}$$

3.4 Model Prediction

After L layers, we obtain the representations of item i and user u at different layers, and then sum them up as the final representation:

$$\mathbf{e}_i = \sum_{l=0}^{L} \mathbf{e}_i^{(l)}, \qquad \tilde{\mathbf{e}}_i = \sum_{l=0}^{L} \tilde{\mathbf{e}}_i^{(l)}, \qquad \mathbf{e}_u = \sum_{l=0}^{L} \mathbf{e}_u^{(l)} \tag{15}$$

By doing so, the complementary information of collaboration and knowledge is separately encoded in the final representations. Different from previous work, we employ the consine similarity to separately predict how likely the user u_i would adopt an item i from item' behavior and attributes aspects, and then adopt the sum of the two predictions as the final prediction score $\hat{y}_{ui}$:

$$\hat{y}_{ui} = \cos(e_u, \tilde{e}_i) + \cos(e_u, e_i) \tag{16}$$

3.5 Model Optimization

We opt for the contrastive loss [31] to optimize HAKG. Compared with widely used BPR loss [33], it alleviates the convergence problem by introducing more negative samples and penalizing uninformative ones. Specifically, for each interaction (u, i) captured

by user-item graph (*i.e.,* positive pair), we randomly sample $|\mathcal{M}_u|$ unobserved items to form the negative pairs together with user u, denoted as $\mathcal{M}_u$, and maximize the similarity between the positive pair and meanwhile minimize the similarity of negative pairs with a margin m:

$$\mathcal{L} = \sum_{(u,i) \in \mathcal{D}} \left[2 - \hat{y}_{ui} + \frac{1}{|\mathcal{M}_u|} \sum_{j \in \mathcal{M}_u} (\hat{y}_{uj} - m)_+ \right] \quad (17)$$

where $\mathcal{D}$ is the interaction data, and $(\cdot)_+$ is the ramp function $\max(0, \cdot)$. By combing the prediction loss and angle loss, we minimize the following objective function to learn the model parameters:

$$\mathcal{L} = \mathcal{L} + \lambda \mathcal{L}'_{\mathrm{H}} \quad (18)$$

where λ is a hyperparameter to control the weight of angle loss defined in Eq. (9).

3.6 Model Analysis

3.6.1 ***Model Size.*** Previous studies [44] have confirmed that discarding the nonlinear feature transformations not only yields better performance but also reduces redundant parameters. Hence, in the aggregation schemes of HAKG, we discard the nonlinear activation function and the learnable transformation metrics. The model parameters of HAKG consist of (1) ID embedding of users, items, KG entities and relations $\{\mathbf{e}_u^{(0)}, \tilde{\mathbf{e}}_i^{(0)}, \mathbf{e}_e^{(0)}, \mathbf{e}_r | u \in \mathcal{U}, i \in \mathcal{I}, e \in \mathcal{E}, r \in \mathcal{R}\}$; and (2) transformation parameters W_1, W_2 for information gated aggregation used in Eq. (12).

3.6.2 ***Time Complexity.*** The time cost of HAKG mainly comes from the aggregation schemes. In the aggregations over KG, the computational complexity of updating knowledge item embeddings in hyperbolic space is $O(|\mathcal{G}_k|dL)$, where $|\mathcal{G}_k|$, d, and L denote the number of KG triplets, the embeddings size, and the number of layers. In the aggregation over user-item graph, the computational complexity of collaborative item embeddings and user embeddings is $O(|\mathcal{G}_b|dL)$, where $|\mathcal{G}_b|$ is the number of interactions. For the hierarchy-aware modeling, the cost of angle constraint is $O(|\mathcal{I}|)$, where $|\mathcal{I}|$ is the number of items. Besides, the cost of mappings between Euclidean space and hyperbolic space is $O(|\mathcal{G}_b| + |\mathcal{G}_k|)$. Thus, the time complexity of the whole training epoch is $O(|\mathcal{G}_k|dL + |\mathcal{G}_b|dL)$. Under the same experimental settings (*i.e.,* same embeddings size), HAKG has comparable complexity to KGAT, CKAN, and KGIN, three representative propagation-based methods.

4 EXPERIMENTS

We present empirical results to demonstrate the effectiveness of our proposed HAKG framework. The experiments are designed to answer the following three research questions:

- **RQ1:** How does HAKG perform, compared with the state-of-the-art knowledge-aware recommendation models?
- **RQ2:** How do different components of HAKG (*i.e.,* hierarchical modeling, gated propagation, dual embeddings, and the depth of propagation layers) affect the performance of HAKG?
- **RQ3:** Can HAKG provide meaningful insights on the hierarchical modelling both from structures and relations?

Table 1: Statistics of the datasets used in our experiments.

		Alibaba-iFashion	Yelp2018	Last-FM
User-Item Interaction	#Users	114,737	45,919	23,566
	#Items	30,040	45,538	48,123
	#Interactions	1,781,093	1,185,068	3,034,796
Knowledge Graph	#Entities	59,156	90,961	58,266
	#H-Relations	50	25	5
	#NH-Relations	1	17	4
	#Triplets	279,155	1,853,704	464,567

4.1 Experimental Settings

4.1.1 ***Dataset Description.*** We utilize three benchmark datasets to evaluate the performance of HAKG: Alibaba-iFashion, Yelp2018, and Last-FM. These three datasets are widely adopted in the state-of-the-art methods [29, 42, 44], and vary in terms of domain, size, and sparsity.

- **Alibaba-iFashion [8].** This is a fashion outfit dataset collected from Alibaba online shopping system, where the outfits (*e.g.,* shoes, tops) are viewed as the items. We follow the work [44] to extract the attributes as the KG data.
- **Yelp2018**[2]**.** This is a local business rating dataset collected by Yelp. We use the 2018 edition dataset of the Yelp challenge, where local businesses like restaurants and bars are viewed as the items.
- **Last-FM**[3]**.** This is a music listening dataset collected from Last.fm music website, where the tracks are viewed as the items. We take the subset of the dataset where the timestamp is from Jan, 2015 to June, 2015.

In order to ensure the data quality, we adopt the 10-core setting [42], *i.e.,* retaining users and items with at least ten interactions and filtering out KG entities involved in less than ten triplets. We use the same data partition with previous studies [42, 44] for comparison (*i.e.,* the proportions of training, validation, and testing set are 80%, 10%, and 10% for all datasets). Besides, we categorize KG relations into hierarchical and non-hierarchical (denoted as H-Relations and NH-Relations). Table 1 summarizes the overall statistics of the three datasets used in our experiments.

4.1.2 ***Evaluation Metrics.*** We adopt the all-ranking strategy [26, 44] to evaluate the performance. Specifically, we treat all the items that user has not adopted before as negative, and treat the relevant items in the testing set as positive. To evaluate the effectiveness of top-K recommendation, we adopt two widely-used evaluation protocols [16, 26] recall@K and ndcg@K, where $K = 20$ by default. We report the average metrics for all the users in the testing set.

4.1.3 ***Baselines.*** In order to demonstrate the effectiveness of HAKG, we compare it with the state-of-the-art methods, including KG-free method (MF), embedding-based methods (CKE and UGRec), propagation-based methods (KGNN-LS, KGAT, CKAN, and KGIN), and hyperbolic-based methods (Hyper-Know and LKGR):

- **MF [33]** is a benchmark factorization model, which only considers the user-item interactions, leaving KG untouched. Here, we use ID embeddings of users and items to perform the prediction.
- **CKE [49]** is a representative embedding method, which utilizes TransR [27] to learn item structural representations from KG, and feeds learned embeddings to MF in an integrated framework.

---

[2]https://www.yelp.com/dataset
[3]https://grouplens.org/datasets/hetrec-2011

- **UGRec** [51] is a state-of-the-art embedding method that models directed and undirected relations from KG and co-occurrence behavior data. While such undirected relations are inaccessible for other methods and unavailable for the three datasets, we add the connectivities between items which are co-interacted by the same user, and treat them as the co-occurrence relationships.
- **KGNN-LS** [40] is a propagation-based model, which transforms KG into user-specific graphs, and then considers user preference on KG relations and label smoothness in the information aggregation phase, so as to generate personalized item representations.
- **KGAT** [42] is a propagation-based recommender. It applies a unified relation-aware attentive aggregation mechanism in UKG to generate user and item representations.
- **CKAN** [46] is based on KGNN-LS, which utilizes different aggregation schemes on the user-item graph and KG respectively, to encode knowledge association and collaborative signals.
- **KGIN** [44] is a state-of-the-art propagation-based method, which models user interaction behaviors with latent intents, and proposes a relation-aware information aggregation scheme to capture long-range connectivity in KG.
- **Hyper-Know** [29] is a state-of-the-art hyperbolic method that embeds KG in hyperbolic space. An adaptive regularization mechanism is also proposed to adaptively regularize items and their neighboring representations.
- **LKGR** [9] is a state-of-the-art hyperbolic GNN method with Lorentz model, which employs different information propagation strategies in the hyperbolic space to encode heterogeneous information from historical interaction and KG.

4.1.4 ***Parameter Settings.*** We implement HAKG in Pytorch, and will release our implementations (codes, parameter settings, and training logs) to facilitate reproducibility. For a fair comparison, we fix the size of ID embeddings as 64, the optimizer as Adam [22], and the batch size as 4096 for all methods. We apply a grid search for hyperparameters: the learning rate is tuned amongst $\{10^{-4}, 10^{-3}, 10^{-2}\}$, the weight of angle loss λ is searched in $\{10^{-5}, 10^{-4}, \cdots, 10^{-2}, 10^{-1}\}$, and tune the GNN layers L in $\{1, 2, 3\}$ for propagation-based methods. For the number of negative samples $|\mathcal{M}_u|$ per user and the margin m of contrastive loss, we set to $\{200, 400, 400\}$ and $\{0.6, 0.8, 0.7\}$ for Alibaba-iFashion, Yelp2018, and Last-FM datasets, respectively. Besides, since optimization in hyperbolic space is practically challenging, we instead define all parameters in the tangent space at the origin, optimize embeddings using standard Euclidean techniques, and use the exponential map to recover the hyperbolic parameters [6]. The parameters for all baseline methods are initialized as those in their original papers, and are carefully tuned to achieve optimal performance. Specifically, for KGAT, we set the depth to three with the hidden size $\{64, 32, 16\}$, and use the pre-trained ID embeddings of MF as initialization; for UGRec, we set the margin value for hinge loss to 1.5; for CKAN, KGNN-LS, and LKGR, we set the size of neighborhood to 16; for KGIN, we fix the number of user intents to 4; and for Hyper-know, we set the curvature to -1.

4.2 Performance Comparison (RQ1)

We begin with the performance comparison *w.r.t.* recall@20 and ndcg@20. The experimental results are reported in Table 2, where %Imp. denotes the relative improvement of the best performing method (starred) over the strongest baselines (underlined). We have the following observations:

- HAKG consistently yields the best performance on all the datasets. In particular, it achieves significant improvement even over the strongest baselines *w.r.t.* ndcg@20 by 15.43%, 8.21%, and 9.79% in Alibaba-iFashion, Yelp2018, and Last-FM, respectively. We attribute these improvement to the hierarchical modeling and holistic information propagation of HAKG: (1) By embedding user-item interactions and KG in hyperbolic space and introducing auxiliary angle constraint for hierarchical relations, HAKG is able to capture the underlying hierarchical structure of data, and better characterize items in the embedding space. In contrast, all baselines ignore the importance of hierarchical relations, and simply use KG-oriented aggregation schemes to capture KG information. (2) Benefited from our gated aggregation with dual item embeddings, HAKG can explicitly preserve the holistic semantics of items to better encode user behavior patterns, while other propagation-based baselines (*e.g.*, CKAN, KGIN, and LKGR) fail to capture the high-order collaborative signals of items.
- Jointly analyzing the performance of HAKG across the three datasets, we find that the improvement on Alibaba-iFashion dataset is more significant than that on other datasets. One possible reason is that the size of KG on Alibaba-iFashion is much smaller than that on Yelp2018 and Last-FM, and most of the relations are hierarchical, thus it is more important to i) preserve the valuable attribute information of items, and ii) capture the crucial collaborative signals of items for learning user preference. This confirms that HAKG can better leverage the user-item interactions and KG for comprehensive item representations.
- The side information of KG is important for recommendations. Compared with vanilla MF, CKE outperforms MF by simply incorporating KG embeddings into MF. The results are consistent with prior studies [44].
- Propagation-based methods outperform embedding methods on most datasets, indicating the importance of modeling long-range connectivities. We attribute their success to the information aggregation schemes. Specifically, recursively collecting information from neighboring nodes is able to capture the high-order latent semantics for high-quality representation.
- Hyperbolic-based methods (*i.e.*, Hyper-know and LKGR) achieve competitive performance compared with Euclidean methods. However, they still slightly underperform state-of-the-art propagation methods (*i.e.*, KGIN). This is because they fail to fully exploit the expressive power of hyperbolic space for comprehensive representations. Specifically, Hyper-know embeds KG in hyperbolic space and directly optimizes the entities' embeddings with triplets, without taking the high-order knowledge associations into consideration. LKGR designs an Euclidean attention mechanism for hyperbolic entities' aggregation, which cannot preserve the local similarities of embeddings in hyperbolic space.

4.3 Study of HAKG (RQ2)

As the hierarchical modeling is at the core of HAKG, we also conduct ablation studies to investigate the effectiveness. Specifically, how the presence of angle loss and gated aggregation, the hyperbolic

Table 2: Overall performance comparison.

	Alibaba-iFashion recall	Alibaba-iFashion ndcg	Yelp2018 recall	Yelp2018 ndcg	Last-FM recall	Last-FM ndcg
MF	0.1095	0.0670	0.0627	0.0413	0.0724	0.0617
CKE	0.1103	0.0676	0.0653	0.0423	0.0732	0.0630
UGRec	0.1006	0.0621	0.0651	0.0419	0.0730	0.0624
KGNN-LS	0.1039	0.0557	0.0671	0.0422	0.0880	0.0642
KGAT	0.1030	0.0627	0.0705	0.0463	0.0873	0.0744
CKAN	0.0970	0.0509	0.0646	0.0441	0.0812	0.0660
KGIN	0.1147	0.0716	0.0698	0.0451	0.0978	0.0848
Hyper-know	0.1057	0.0648	0.0685	0.0447	0.0948	0.0812
LKGR	0.1033	0.0612	0.0679	0.0438	0.0883	0.0675
HAKG	**0.1319***	**0.0848***	**0.0778***	**0.0501***	**0.1008***	**0.0931***
%Imp.	14.99%	15.43%	10.35%	8.21%	3.07%	9.79%

Table 3: Impact of angle loss and gated aggregation.

	Alibaba-iFashion recall	Alibaba-iFashion ndcg	Yelp2018 recall	Yelp2018 ndcg	Last-FM recall	Last-FM ndcg
w/o A&G	0.1218	0.0799	0.0737	0.0458	0.0946	0.0872
w/o A	0.1272	0.0825	0.0763	0.0485	0.0963	0.0907
w/o G	0.1253	0.0817	0.0758	0.0471	0.0959	0.0894

Table 4: Impact of hierarchical modeling.

	Alibaba-iFashion recall	Alibaba-iFashion ndcg	Yelp2018 recall	Yelp2018 ndcg	Last-FM recall	Last-FM ndcg
Euclidean	0.1231	0.0798	0.0756	0.0484	0.0981	0.0916
PH-Relation	0.1317	0.0845	0.0772	0.0494	0.1001	0.0928
AH-Relation	0.1306	0.0837	0.0771	0.0492	0.0995	0.0924

Table 5: Impact of dual item embedding.

	Alibaba-iFashion recall	Alibaba-iFashion ndcg	Yelp2018 recall	Yelp2018 ndcg	Last-FM recall	Last-FM ndcg
Single	0.1186	0.0755	0.0769	0.0492	0.0989	0.0904
Dual	0.1319	0.0847	0.0778	0.0501	0.1008	0.0931

Table 6: Impact of the number of layers L.

	Alibaba-iFashion recall	Alibaba-iFashion ndcg	Yelp2018 recall	Yelp2018 ndcg	Last-FM recall	Last-FM ndcg
HAKG-1	0.1313	0.0845	0.0766	0.0489	0.0972	0.0897
HAKG-2	0.1306	0.0831	0.0778	0.0501	0.0988	0.0913
HAKG-3	0.1319	0.0848	0.0774	0.0498	0.1008	0.0931

order to compensate for the unavailability of relation types (named PH-Relation), and iii) simply assuming that item connected relations are the hierarchical relations and use them to characterize items (termed as AH-Relation). The results of three alternative models are listed in Table 4. We observe that:

- The performance degrades for all three datasets when we remove the hyperbolic geometry for HAKG, meaning that modeling user-item interactions and KG in hyperbolic space could enhance the model's expressive power and yield better representations for recommendation.
- When the hierarchical types of KG relations are not explicitly available, the performance of alternative approaches (*i.e.,* PH-Relation and AH-Relation) approximates to that of original model with ground-truth hierarchical relations, which empirically shows that our model is robust for different datatsets even when the KG's hierarchical information is not available.

4.3.3 **Impact of Dual Item Embeddings.** To analyze the effectiveness of the design of dual item embeddings, we compare it with the conventional single item embeddings, which removes the additional collaborative item embeddings and directly leverages the knowledge embeddings of items in Eq. (7) for users' aggregation. We compare their performance on three datasets, and present the results in Table 5. We have the following findings:

- Discarding the collaborative item embeddings would consistently degrade the performance cross three datasets. This is because if we model items with single representations, they inevitably suffer from information loss when performing the neighbor aggregation. In other words, since items are inherently exhibit two kinds of semantics (collaboration and knowledge), it is unlikely to simultaneously incorporate all the high-order information of items with single embeddings.
- The performance of single item embeddings decrease more dramatically on Aliabab-iFashion dataset, compared with it on other two datasets. One possible reason is that the amount of auxiliary information provided by KG is the smallest, which makes it more important to capture the collaborative signals of items with additional collaborative item embeddings.

4.3.4 **Impact of Model Depth.** We also explore the impact of the number of aggregation layers. Stacking more layers is able to collect the high-order collaborative signals and knowledge associations for better capturing of the latent user behavior patterns but at a
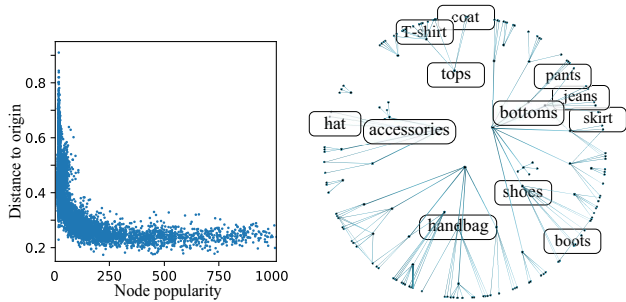
embeddings and hierarchical relations, the dual item embeddings, and the number of propagation layers affect our model.

4.3.1 **Impact of Angle Loss & Gated Aggregation.** We first verify the effectiveness of the angle loss and gated aggregation. To this end, three variants of HAKG are constructed by (1) discarding the angle constraint and gated aggregation scheme, termed as $\mathrm{HAKG_{w/o\ A\&G}}$, (2) removing the angle loss for hierarhical triplets, called $\mathrm{HAKG_{w/o\ A}}$, and (3) replacing the gated aggregation with simple point-wise addition, named $\mathrm{HAKG_{w/o\ G}}$. We summarize the results in Table 3.

Compared with the complete model of HAKG in Table 2, the absence of the angle constraint and gated aggregation dramatically degrades the performance, indicating the necessity of modeling hierarchical relation and collaborative signals. Specifically, $\mathrm{HAKG_{w/o\ A\&G}}$ directly fuses the dual item embeddings for user aggregation, and ignores the hierarchical relations in KG, and thus, it fails to profile item properly and propagate comprehensive information for learning use. Analogously, leaving the hierarchical relations unexplored (*i.e.,* $\mathrm{HAKG_{w/o\ A}}$) also downgrades the performance. Although $\mathrm{HAKG_{w/o\ G}}$ retains the modeling of hierarchical relations for characterizing items, it is unable to provide discriminative signals for identifying user behavior patterns, incurring suboptimal user representations.

4.3.2 **Impact of Hierarchical Modeling.** We then evaluate the influence of hierarchical modeling by considering both hierarchical structure and relations. To be more specific, we propose three alternative models, which are modified by: i) replacing all hyperbolic operations with Euclidean alternatives and retaining the computation logic of HAKG (called Euclidean), ii) predicting the hierarchical types of relations by widely used Krackhardt scores criteria [2, 25] in

Figure 4: (1) Distance from node (user or item) embedding to origin vs node popularity. (2) Hyperbolic embeddings of entities in KG.

higher cost. Here, we search L in the range of $\{1, 2, 3\}$, and report the results in Table 6. We have the following observations:

- Generally speaking, increasing the aggregation layers can enhance the performance, especially for Last-FM datasets. We attribute such improvement to two reasons: (1) Gathering more relevant collaborative signals and knowledge association could provide informative semantics for learning high-quality representations, deepening the understanding of user interest. (2) The dual item embeddings explicitly encode both items' behavior and attribute similarities, which profile item at a finer-grained way than single item embeddings used in other methods.

- It is worth mentioning that our model is less sensitive to the model depth, compared with other propagation-based methods [42, 44, 46]. Specifically, HAKG could achieve competitive performance even when $L = 1$. This is because our separate aggregation schemes and gate mechanism can directly capture useful patterns from both user-item interactions and KG, while the KG-oriented aggregation schemes need more layers to encode the latent semantics from the mixed and obscure information.

4.4 Hierarchies Visualization (RQ3)

In this section, we visualize the learned embeddings to give an intuitive impression of our hierarchical modeling. Specifically, we first train HAKG with two-dimensional embeddings on the Alibaba-iFashion dataset, and separately analyze the hierarchies that exhibit in $\mathcal{G}_b$ and $\mathcal{G}_k$: (1) For the underlying hierarchical structure in interactions, we plot the distance from the origin to hyperbolic embeddings (*e.g.,* user embeddings or collaborative item embeddings) versus its popularity. (2) For the hierarchical relations in KG, we select representative entities and demonstrate their connectivities with triplets. As shown in Figure 4, we find that:

- The left of Figure 4 indicates a clear exponential trend that distance to the origin increases exponentially for less popular items, which is consistent with the degree distribution of interactions in Figure 1(b). This confirms that our model takes advantage of the exponentially growing volume in hyperbolic space, and uses it to naturally represent the users and items.

- The connectivities in the right of Figure 4 show clear hierarchical relations between entities. For instance, the entity *bottoms* is categorized into *pants*, *jeans*, and *skirt*, and the attribute information of items (*e.g., T-shirt* is a type of *tops*) can be naturally preserved with the hierarchical relations. Thus, HAKG is able to capture

the items' attribute semantics in the embedding space, and yield better item representations for learning user preferences.

5 RELATED WORK

Existing recommendation systems incorporated with KG information can be mainly categorized into three clusters, viz., *embedding-based methods*, *path-based methods*, and *propagation-based methods*. We briefly review them in the following.

- **Embedding-based methods** [1, 4, 19, 37, 39, 49, 51] directly embed entities and relations in KG via knowledge graph embedding (KGE) methods (*e.g.,* TransR [27] and TransD [20]) to serve as item embedding in recommendation. For example, CKE [49] utilizes TransR to learn item structural representations from knowledge graph, and feeds the learned embeddings to matrix factorization (MF) [33] in an integrated framework. Hyper-Know [29] embeds knowledge graph in Poincaré Ball and then designs an adaptive regularization mechanism to regularize item representations. Although these methods benefit from the simplicity and expressiveness of KGE, they fail to capture high-order dependence of user-item relations for user preference learning.

- **Path-based methods** [5, 17, 21, 30, 45, 52] aim to find semantic paths in KG, and then connect items and users to discover long-range connectivity for recommendation. Those paths can be used to predict user preference with recurrent neural networks [45, 52] or attention mechanism [17]. For instance, KPRN [45] captures the sequential dependence within a knowledge-aware path to infer the underlying high-order relation of a user-item interaction. However, defining proper meta-path patterns requires domain knowledge, which can be extremely time-consuming for complicated KG with various types of entities and relations. Moreover, domain-specific meta-paths inevitably lead to poor generalization for different recommendation scenarios [21, 28].

- **Propagation-based methods** [34, 38, 40–42, 44, 46] are inspired by the recent advances of graph neural networks (GNNs) [14, 15, 23, 35, 43, 47], which iteratively perform information aggregation mechanism from neighborhood nodes. As such, these methods are able to discover high-order relations in an end-to-end fashion. For example, KGAT [42] creates unified knowledge graph (UKG) to combine user-item interactions and KG, and then performs knowledge-aware attention on it. CKAN [46] employs the heterogeneous propagation strategy to encode the user preference with knowledge associations to further improve the performance. Most recently, KGIN [44] considers a new aggregation mechanism to integrate long-range relation paths, and disentangles user preference with intents for better interpretability.

How to leverage hyperbolic geometry and hyperbolic embeddings in recommender systems has become a recent trend [9, 29, 36, 50]. For instance, HyperML [36] is the first work to leverage hyperbolic geometry for recommendation through metric learning approach; and LKGR [9] proposes a hyperbolic GNN method for knowledge-aware recommendation. However, to the best of our knowledge, none of existing methods considers modeling KG relations at a finer-grained level of hierarchies. Moreover, they fail to preserve the crucial high-order collaborative signals of items. Our work differs from them in hierarchical modeling and aggregation, i.e., we aim at addressing the importance of hierarchical

relations for profiling items, and explicitly propagate two kinds of information with dual item embeddings for better identifying user behaviors.

6 CONCLUSION AND FUTURE WORK

In this paper, we propose a new knowledge-aware recommendation model, called HAKG, which is able to capture the underlying hierarchical structure of data in hyperbolic space, and characterize items with hierarchical relations in KG. Besides, HAKG employs a dual item embeddings design to separately encode items' collaborative signals and knowledge associations, and develops a gated mechanism to control discriminative signals towards the users' behavior patterns. Extensive experiments conducted on three real-world benchmark datasets demonstrate the superiority of HAKG. In the near future, we plan to investigate the evolving process of user interests with KG, to dynamically model the users' activated core interests with the long-term behavior sequences and meanwhile examine all aggregation and propagation operations (*e.g.,* addition and multiplication) in hyperbolic space to fully exploit the power of hyperbolic geometry.

ACKNOWLEDGMENTS

REFERENCES

[1] Qingyao Ai, Vahid Azizi, Xu Chen, and Yongfeng Zhang. 2018. Learning Heterogeneous Knowledge Base Embeddings for Explainable Recommendation. *Algorithms* 11, 9 (2018), 137–146.

[2] Yushi Bai, Rex Ying, Hongyu Ren, and Jure Leskovec. 2021. Modeling Heterogeneous Hierarchies with Relation-specific Hyperbolic Cones. In *NeurIPS*.

[3] James W. Cannon, William J. Floyd, Richard Kenyon, Walter, and R. Parry. 1997. Hyperbolic geometry. In *In Flavors of geometry.* 59–115.

[4] Yixin Cao, Xiang Wang, Xiangnan He, Zikun hu, and Tat-Seng Chua. 2019. Unifying Knowledge Graph Learning and Recommendation: Towards a Better Understanding of User Preferences. In *WWW*. 151–161.

[5] Rose Catherine and William Cohen. 2016. Personalized Recommendations Using Knowledge Graphs: A Probabilistic Logic Programming Approach. In *RecSys*. 325–332.

[6] Ines Chami, Adva Wolf, Da-Cheng Juan, Frederic Sala, Sujith Ravi, and Christopher Ré. 2020. Low-Dimensional Hyperbolic Knowledge Graph Embeddings. In *ACL*. 6901–6914.

[7] Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. 2019. Hyperbolic graph convolutional neural networks. (2019), 4868–4879.

[8] Wen Chen, Pipei Huang, Jiaming Xu, Xin Guo, Cheng Guo, Fei Sun, Chao Li, Andreas Pfadler, Huan Zhao, and Binqiang Zhao. 2019. POG: personalized outfit generation for fashion recommendation at Alibaba iFashion. In *KDD*. 2662–2670.

[9] Yankai Chen, Menglin Yang, Yingxue Zhang, Mengchen Zhao, Ziqiao Meng, Jian Hao, and Irwin King. 2022. Modeling Scale-free Graphs for Knowledge-aware Recommendation. In *WSDM*.

[10] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

[11] Maurice Fréchet. 1948. Les éléments aléatoires de nature quelconque dans un espace distancié. In *Annales de l'institut Henri Poincaré*. 215–310.

[12] Octavian Ganea, Gary Becigneul, and Thomas Hofmann. 2018. Hyperbolic Neural Networks. In *NeurIPS*. 5350–5360.

[13] Octavian-Eugen Ganea, Gary Bécigneul, and Thomas Hofmann. 2018. Hyperbolic Entailment Cones for Learning Hierarchical Embeddings. In *ICML*. 1646–1655.

[14] William L. Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *NeurIPS*. 1025–1035.

[15] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, YongDong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *SIGIR*. 639–648.

[16] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In *WWW*. 173–182.

[17] Binbin Hu, Chuan Shi, Wayne Xin Zhao, and Philip S. Yu. 2018. Leveraging Meta-Path Based Context for Top- N Recommendation with A Neural Co-Attention Model. In *KDD*. 1531–1540.

[18] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative Filtering for Implicit Feedback Datasets. In *ICDM*. 263–272.

[19] Jin Huang, Wayne Xin Zhao, Hongjian Dou, Ji-Rong Wen, and Edward Y. Chang. 2018. Improving Sequential Recommendation with Knowledge-Enhanced Memory Networks. In *SIGIR*. 505–514.

[20] Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Knowledge Graph Embedding via Dynamic Mapping Matrix. In *ACL*. 687–696.

[21] Jiarui Jin, Jiarui Qin, Yuchen Fang, Kounianhua Du, Weinan Zhang, Yong Yu, Zheng Zhang, and Alexander J. Smola. 2020. An Efficient Neighborhood-Based Interaction Model for Recommendation on Heterogeneous Graph. In *KDD*. 75–84.

[22] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. In *ICLR*.

[23] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *ICLR*.

[24] Yehuda Koren. 2008. Factorization Meets the Neighborhood: A Multifaceted Collaborative Filtering Model. In *KDD*. 426–434.

[25] David Krackhardt. 2014. Graph theoretical dimensions of informal organizations. In *Computational organization theory*. 107–130.

[26] Walid Krichene and Steffen Rendle. 2020. On sampled metrics for item recommendation. In *KDD*. 1748–1757.

[27] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning Entity and Relation Embeddings for Knowledge Graph Completion. In *AAAI*. 2181–2187.

[28] Yuanfu Lu, Yuan Fang, and Chuan Shi. 2020. Meta-Learning on Heterogeneous Information Networks for Cold-Start Recommendation. In *KDD*. 1563–1573.

[29] Chen Ma, Liheng Ma, Yingxue Zhang, Haolun Wu, Xue Liu, and Mark Coates. 2021. Knowledge-Enhanced Top-K Recommendation in Poincaré Ball. In *AAAI*. 4285–4293.

[30] Weizhi Ma, Min Zhang, Yue Cao, Woojeong Jin, Chenyang Wang, Yiqun Liu, Shaoping Ma, and Xiang Ren. 2019. Jointly Learning Explainable Rules for Recommendation with Knowledge Graph. In *WWW*. 1210–1221.

[31] Kelong Mao, Jieming Zhu, Jinpeng Wang, Quanyu Dai, Zhenhua Dong, Xi Xiao, and Xiuqiang He. 2021. SimpleX: A Simple and Strong Baseline for Collaborative Filtering. In *CIKM*. 1243–1252.

[32] Maximillian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In *NeurIPS*. 6338–6347.

[33] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *UAI*. 452–461.

[34] Ke Tu, Peng Cui, Daixin Wang, Zhiqiang Zhang, Jun Zhou, Yuan Qi, and Wenwu Zhu. 2021. Conditional Graph Attention Networks for Distilling and Refining Knowledge Graphs in Recommendation. In *CIKM*. 1834–1843.

[35] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *ICLR*.

[36] Lucas Vinh Tran, Yi Tay, Shuai Zhang, Gao Cong, and Xiaoli Li. 2020. Hyperml: A boosting metric learning approach in hyperbolic space for recommender systems. In *WSDM*. 609–617.

[37] Chenyang Wang, Min Zhang, Weizhi Ma, Yiqun Liu, and Shaoping Ma. 2020. Make It a Chorus: Knowledge- and Time-Aware Item Modeling for Sequential Recommendation. In *SIGIR*. 109–118.

[38] Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. 2018. RippleNet: Propagating User Preferences on the Knowledge Graph for Recommender Systems. In *CIKM*. 417–426.

[39] Hongwei Wang, Fuzheng Zhang, Xing Xie, and Minyi Guo. 2018. DKN: Deep knowledge-aware network for news recommendation. In *WWW*. 1835–1844.

[40] Hongwei Wang, Fuzheng Zhang, Mengdi Zhang, Jure Leskovec, Miao Zhao, Wenjie Li, and Zhongyuan Wang. 2019. Knowledge-Aware Graph Neural Networks with Label Smoothness Regularization for Recommender Systems. In *KDD*. 968–977.

[41] Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. 2019. Knowledge Graph Convolutional Networks for Recommender Systems. In *WWW*. 3307–3313.

[42] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. KGAT: Knowledge Graph Attention Network for Recommendation. In *KDD*. 950–958.

[43] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In *SIGIR*. 165–174.

[44] Xiang Wang, Tinglin Huang, Dingxian Wang, Yancheng Yuan, Zhenguang Liu, Xiangnan He, and Tat-Seng Chua. 2021. Learning Intents behind Interactions with Knowledge Graph for Recommendation. In *WWW*. 878–887.

[45] Xiang Wang, Dingxian Wang, Canran Xu, Xiangnan He, Yixin Cao, and Tat-Seng Chua. 2019. Explainable Reasoning over Knowledge Graphs for Recommendation. In *AAAI*. 5329–5336.

[46] Ze Wang, Guangyan Lin, Huobin Tan, Qinghong Chen, and Xiyang Liu. 2020. CKAN: Collaborative Knowledge-Aware Attentive Network for Recommender Systems. In *SIGIR*. 219–228.

[47] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying Graph Convolutional Networks. In *ICML*. 6861–6871.

[48] Yikun Xian, Zuohui Fu, Shan Muthukrishnan, Gerard De Melo, and Yongfeng Zhang. 2019. Reinforcement knowledge graph reasoning for explainable recommendation. In *SIGIR*. 285–294.

[49] Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. 2016. Collaborative Knowledge Base Embedding for Recommender Systems. In *KDD*. 353–362.

[50] Sixiao Zhang, Hongxu Chen, Xiao Ming, Lizhen Cui, Hongzhi Yin, and Guandong Xu. 2021. Where Are We in Embedding Spaces? A Comprehensive Analysis on Network Embedding Approaches for Recommender Systems. In *KDD*. 2223–2231.

[51] Xinxiao Zhao, Zhiyong Cheng, Lei Zhu, Jiecai Zheng, and Xueqing Li. 2021. UGRec: Modeling Directed and Undirected Relations for Recommendation. In *SIGIR*. 193–202.

[52] Sun Zhu, Yang Jie, Zhang Jie, Bozzon Alessandro, Huang Long-Kai, and Xu Chi. 2018. Recurrent Knowledge Graph Embedding for Effective Recommendation. In *RecSys*. 297–305.