Decoupled Hyperbolic Graph Attention Network for Modeling Substitutable and Complementary Item Relationships

Anonymous Author(s)

Abstract

Modeling substitutable and complementary item relationships is a fundamental and important topic for recommendation in e-commerce online scenarios. In the real world, item relationships are usually coupled, heterogeneous and they also have abundant side information and hierarchical data structures. Recently, to take full advantage of both sides information and topological structure, graph neural networks are widely explored in relationship modeling. However, the existing methods are crude in decoupling heterogeneous relationships. Their model designs lack deep insight of relationships' coupling mode, i.e. neglects the prior knowledge of how relationships affect each other. In addition, many existing graph methods, regardless of how they handle coupled relationships, are deployed in Euclidean spaces, which distorts hierarchical data structure and limits the expressive power due to the non power law characteristic of Euclidean topology.In this paper, we propose a novel **D**ecoupled **H**yperbolic **G**raph **A**ttention **N**etwork (DHGAN). The innovations of our DHGAN can be highlighted as two aspects. Firstly, we design metapaths in an adequate way following an algebraic perspective of relationships coupling mode, which helps achieving better model interpretability. Secondly, DHGAN maps heterogeneous relationships into separate hyperbolic spaces, which can better capture the hierarchical information of graph nodes and helps improving model's representational capacity. We conduct extensive experiments on three public real-world datasets, demonstrating DHGAN is superior to the state-of-the-art graph baselines.

CCS Concepts

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

Keywords

substitutable and complementary item relationships, graph neural networks, hyperbolic space, algebraic perspective

ACM Reference Format:

Anonymous Author(s). 2022. Decoupled Hyperbolic Graph Attention Network for Modeling Substitutable and Complementary Item Relationships. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'CIKM)*. ACM, New York, NY, USA, 12 pages. https://doi.org/XXXXXXX.XXXXXXX

Conference acronym 'CIKM, XX 03–05, 2022, XXXXX, XX

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Over the past decades, Recommender Systems have grown significantly in a variety of scenarios, such as e-commerce, social media, video and online news platforms. For e-commerce website, the primary task of a recommender system is to provide proper items (i.e. products) that user might like. Therefore, the ability to accurately simulate user preferences and item characteristics is crucial to guarantee the quality of a recommender system.

With the rapid proliferation of e-commerce platforms (e.g., Amazon, JD.com, Taobao), rich information about various kinds of items has become available. These enormous items are explicitly or implicitly connected via different types of relationships, e.g., co-viewed or co-purchased by the same user. Multiple user behaviours help us form massive item graphs, where the items are represented as nodes and their relationships are represented as edges. An item graph with multiple relationships is a typical heterogeneous graph. In real-world recommendation scenarios, graph networks with such prototypical characteristics of complex structures are widespread, and are generally power law distribution [16, 28, 32].

Among these multiplex item relationships, substitutable and complementary relationships are of great significance when recommended to users with different purchase intentions [23, 45] . Normally speaking, substitutes are products that can be purchased instead of each other, while complements are products that can be purchased in addition to each other. The substitutable and complementary items of users' recently interacted products are most important sources in generating new personalized candidate items. How can we capture substitutable and complementary relationships accurately is an essential problem in modern recommender systems.

Many approaches have explored this problem, and achieved some meaningful results [13, 24, 31, 43, 48–50, 52]. However, a majority of them only focus on the attributes information from item itself (e.g., visual and textual content) while lacking of exploration to the rich information of heterogeneous graph structure, such as generative model Latent Dirichlet Allocation (LDA) [24], Linked Variational Autoencoder (LVA) [31], neural network model ENCORE [50] and RRN [48].

As we mentioned before, multiple item relationships actually form a heterogeneous item graph. Recent achievements of Network Embeddings and Graph Neural Networks provide more possibilities in relationship prediction problems, and several studies are inspired to borrow graph techniques to capture item attributes as well as graph topology [19, 43, 47, 49]. It is worth mentioning that Liu et al. propose a graph convolutional network method that decouples substitutable and complementary relationships, as well as considers mutual influence of each other.

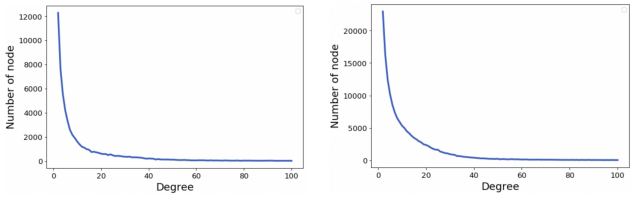
Although the previous GNN-based models have made leading performance, the exploration of heterogeneous topological information of item relationships is still under-exploited. To be more

specific, most of the existing GNN-based methods suffer from two shortcomings.

1) **Lack of model interpretability.** The existing GNN-based relationships decoupling methods are sort of too crude when dealing with the heterogeneous data. Although some models incorporate the mutual influence between different relationships, they are more like direct mixing of relationships via attention mechanisms, without considering how relationships interact with each other. In fact, fully understanding and exploiting the coupling pattern of heterogeneous item relationships, which act as a kind of priori knowledge, can help us understand how relationships interact with each other and help design more effective decoupling models.
2) **Distort hierarchical data structure.** Most existing GNN-based relationships prediction methods learn item representations in Euclidean spaces. However, Barabási and Bonabeau found that most real networks are scale free networks with hierarchical structure (including item relationship networks) which the distribution followed a so-called power law [28, 32]. Figure 1 shows the degree distribution of the experimental data used in this paper, which validates the theory in [2], i.e., the substitutable and complementary relationships graph appears power law distribution.
   If we model the power law distribution data in an Euclidean space, for example, in a two-dimensional Euclidean space, the number of nodes increases polynomial from the center to the radius, instead of an expected exponential rate [16]. That is to say, Euclidean mechanism of graph model will distort data hierarchy, and increase the difficulty of capturing rich heterogeneous information contained in item graph data due to the limitation of representational capacity.

To address the above shortcomings, in this paper we propose a novel graph neural network framework to infer substitutable and complementary relationships, namely **D**ecoupled **H**yperbolic **G**raph **A**ttention **N**etwork (**DHGAN**). Specifically, to address the first limitation, we first analyze the coupling mode between substitutable and complementary relationships in an algebraic perspective, and extract valuable coupling patterns from the heterogeneous graph. Based on those patterns, we design concise and reasonable metapaths using only one-hop neighbors of the target node. An attention mechanism based on the hyperbolic proximity is applied to aggregate messages along designed metapaths. As for the second limitation, our model maps item representations into hyperbolic spaces, which can keep the node growth rate from center to the radius in an exponential-level, indicating a stronger representation ability than Euclidean space [1, 16, 42]. The central node of the scale free network that satisfies the power law distribution, which increases the number of neighbors exponentially with the increase of hop. Therefore, compared with Euclidean space, hyperbolic space is more suitable as the embedding space of scale free network. Our model is effectively learned by cross-entropy loss based on hyperbolic distances optimized with the Riemannian Gradient Descent.

In sum, the major contributions of this work can be summarized as follows:



(a) Substitutable relationship graph (b) Complemenary relationship graph

Figure 1: Degree distribution for substitutable relationship graph and complemenary relationship graph.

- To the best of our knowledge, we are the first to introduce hyperbolic representation learning to multiple item relationships learning.
- We explore the coupling mode between substitutable and complementary relationships, and design meaningful novel metapaths to aggregate decoupling messages only using one-hop neighbors. Heterogeneous graph information is leveraged by a hyperbolic attention mechanism.
- We conduct extensive experiments on three public datasets demonstrating superior performance over the state-of-the-art baselines.

2 Related Work

2.1 The Inference of Item Relationships

For modern E-commerce platforms, understanding how products relate to each other, such as typical substitute and complement relationships, is of great importance to provide convenience for users and boost extra sales for platforms.

In the past decade, extensive studies formulate this item relationship inference problem as a link prediction problem [24, 31, 43, 50]. The majority of these studies focus on perceiving different content features of items [24, 31], such as visual and textual content, while neglecting the effect of graph structure.

With the rapid development of Graph Embedding [8, 11, 21, 29, 35] and Graph Neural Networks [12, 15, 39, 46, 53], a bunch of studies start introducing graph techniques into relationships learning. For example, Ying et al. is the first to introduce Graph Neural Networks to infer substitutable items, and propose a unified modeling method for both item features and local graph structures.

Most existing methods, using graph structures or not, fuse different relationships together into a single shared item representation [4, 24]. However, due to implicit or obvious multifaceted item semantics, one single representation might be non-robust and lack of interpretability [9, 17]. Notably, Ma et al. proposes DisenGCN to decouple node representations in Graph Neural Networks. Wang et al. introduces this idea into recommender systems and learns multiple embeddings w.r.t. users' possible intents.

From the perspective of item relationships, these different relationships are not separated completely. In fact, they are interacted naturally. For example, two substitutable computers are likely to share many common complementary items (e.g. keyboards). Liu et al. designs integration processes consists of structural integration and semantic integration to consider the mutual influence when learning decoupled representations, which is inspiring but still lack

of convincing logical interpretation. Their structural integration mechanism actually equals to passing messages along designed metapath which is a rough arrangement of one-hop neighbors of different types. Their semantic integration fuses representations of the same node via knowledge transfer techniques. However, one embedding of a single node is meaningless, and the node embedding is meaningful only when it is with other nodes' embeddings, i.e.,the binary pair relationship is valuable. Therefore, the benefit of fusing embeddings from different spaces might be very small.

2.2 Hyperbolic Embedding Learning

All the methods introduced above, map their final embeddings into Euclidean spaces, and the calculations therein are naturally designed in the Euclidean fashion. Although Euclidean-based models have achieved outstanding performance in many fields, the non-power law property limits their representation abilities on data with hierarchical structure [1, 42], such as text and graph data.

To tackle this problem, Nickel and Kiela consider learning nodes representations in a hyperbolic space. They have demonstrated that the hyperbolic spaces could be the latent spaces of graph data, as the hyperbolic space may maintain some properties of graph naturally, e.g., hierarchical with power law distribution. Thereafter many improved approaches are developed [6, 18, 34, 51], Chami et al. propose Hyperbolic Graph Convolutional Neural Network (HGCN) which is first inductive hyperbolic GCN, HGCN achieves the experiment results superior to state-of-the-art GCNs in Pubmed dataset. Liu et al. propose a novel GNN architecture for learning representations on Riemannian manifolds with differentiable exponential and logarithmic maps, the model can lead to substantial improvements on various benchmark datasets. Zhang et al. study the GNN with attention mechanism in hyperbolic spaces at the first attempt, the comprehensive experimental results on four real-world datasets demonstrate the performance of the hyperbolic graph attention network model. Recently hyperbolic embedding techniques have been applied to a variety of problems in different areas, such as Natural Language Processing [7] and Computer Vision [14].

Due to the hierarchical data structures in many recommendation scenarios, hyperbolic embedding technique has also been rapidly popularized in Recommender System [5, 10, 25, 33]. For examples, Mirvakhabova et al. introduces a single layer Autoencoder based model on hyperbolic geometry in the top-N recommendation problem, Feng et al. propose a hyperbolic metric embedding model for Next Point-of-Interest recommendation, Chamberlain et al. propose an asymmetric hyperbolic collaborative filtering method using the Einstein midpoint and a weighted margin-rank batch loss, and Sun et al. apply hyperbolic Graph Convolutional Networks in collaborative filtering problem.

3 Preliminaries

In this section, we first recall some necessary definitions and concepts in hyperbolic geometry, and then formulate the task of heterogeneous item relationships learning.

3.1 Gyrovector Spaces

First, we introduce general definitions of Riemannian manifold and hyperbolic space. Then we describe a specific hyperbolic space, i.e., Gyrovector space, which has some good operational properties.

A Riemannian manifold $(\mathcal{M}, g)$ is a real, smooth manifold $\mathcal{M}$ equipped with a Riemannian metric g which deduces a positive-definite inner product $g_x : \mathcal{T}_x\mathcal{M} \times \mathcal{T}_x\mathcal{M} \to \mathbb{R}$ on the tangent space $\mathcal{T}_x\mathcal{M}$ at each point x. The curvature of a Riemannian manifold, intuitively, is the amount by which a curve deviates from being a straight line, or a surface deviates from being a plane. A hyperbolic space is a Riemannian manifold that has a constant negative curvature. For more details, please refer to [30].

The Gyrovector space is a hyperbolic space $(\mathbb{D}_k^d, g^k)$, where $\mathbb{D}_k^d = \{x \in \mathbb{R}^d : k\|x\| < 1, k \geqslant 0\}$. If we denote its curvature by q, then $k = -\frac{1}{q}$, and $g_x^k = \lambda_x^2 g^E$ at point $x \in \mathbb{D}_k^d$ with $\lambda_x := \frac{2}{1-k\|x\|^2}$ being a transformation related to the Euclidean norm and $g^E = \mathbf{I}_d$ being the Euclidean metric tensor.

As in Euclidean space, operations in Gyrovector space are inherited from the vectorial structure. Then we can define basic vector operations in Gyrovector space, such as vector addition, scalar multiplication and matrix multiplication [36–38]. Some typical operation definitions are as follows.

Möbius addition. *$\forall x, y \in \mathbb{D}_k^d$, the Möbius addition $\oplus_k$ is given by:*

$$x \oplus_k y := \frac{(1 + 2k\langle x, y\rangle + k\|y\|^2)x + (1 - k\|x\|^2)y}{1 + 2k\langle x, y\rangle + k^2\|x\|^2\|y\|^2}.$$

Möbius scalar multiplication. *For $k > 0$, $c \in \mathbb{R}$ and $x \in \mathbb{D}_k^d \setminus \{\mathbf{0}\}$, the Möbius scalar multiplication $\odot_k$ is given by:*

$$c \odot_k x := \frac{1}{\sqrt{k}} \tanh(c \cdot \tanh^{-1}(\sqrt{k}\|x\|)) \frac{x}{\|x\|}.$$

Möbius matrix multiplication. *For $k > 0$, $M \in \mathbb{R}^{b\times d}$ and $x \in \mathbb{D}_k^d$. If $Mx \neq 0$, the Möbius matrix multiplication is given by:*

$$M \otimes_k x := \frac{1}{\sqrt{k}} \tanh\left(\frac{\|Mx\|}{\|x\|} \tanh^{-1}(\sqrt{k}\|x\|)\right) \frac{Mx}{\|Mx\|},$$

where $M \otimes_k x \in \mathbb{D}_k^b$. If $Mx = 0$, $M \otimes_k x := 0$.

Distance. The distance function d_k of any pair of points $x, y \in \mathbb{D}_k^d$ is given by:

$$d_k(x, y) = \frac{2}{\sqrt{k}} \tanh^{-1}(\sqrt{k}\| - x \oplus_k y\|).$$

Exponential and logarithmic maps. For $x \in \mathbb{D}_k^d$, $y \in \mathbb{D}_k^d$ and $v \in \mathcal{T}_x\mathbb{D}_k^d$ where $v \neq 0$ and $y \neq x$, the exponential map $\exp_x^k : \mathcal{T}_x\mathbb{D}_k^d \to \mathbb{D}_k^d$ and logarithmic map $\log_x^k : \mathbb{D}_k^d \to \mathcal{T}_x\mathbb{D}_k^d$ are given by:

$$\exp_x^k(v) := x \oplus_k \left(\tanh\left(\sqrt{k}\frac{\lambda_x^k\|v\|}{2}\right)\frac{v}{\sqrt{k}\|v\|}\right),$$

$$\log_x^k(y) := \frac{2}{\sqrt{k}\lambda_x^k} \tanh^{-1}(\sqrt{k}\| - x \oplus_k y\|) \frac{-x \oplus_k y}{\| - x \oplus_k y\|}.$$

This closed-form expression connects hyperbolic space with tangent space. When $x = \mathbf{0}$, two maps are defined as:

$$\exp_{\mathbf{0}}^k(v) = \tanh(\sqrt{k}\|v\|)\frac{v}{\sqrt{k}\|v\|},$$

$$\log_{\mathbf{0}}^k(y) = \tanh^{-1}(\sqrt{k}\|y\|)\frac{y}{\sqrt{k}\|y\|}.$$

3.2 Heterogeneous Item Relationships Inference

Multiple item relationships generate a heterogeneous graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{X}, \psi)$, where $\mathcal{V}$ denotes the set of n items, $\mathcal{E}$ represents the set of edges, $\mathcal{X}$ is the set of nodes' attributes, and ψ maps each edge into the set of relationships $\mathcal{R}$.

If we only consider substitutable and complementary relationships, we can represent the set of relationships as $\mathcal{R} = \{s, c\}$, and $\mathcal{E}$ as $\mathcal{E}^{(s)} \cup \mathcal{E}^{(c)}$. Then two subgraphs of $\mathcal{G}$ are obtained: substitutable item subgraph $\mathcal{G}^{(s)} = (\mathcal{V}, \mathcal{E}^{(s)}, \mathcal{X})$ and complementary item subgraph $\mathcal{G}^{(c)} = (\mathcal{V}, \mathcal{E}^{(c)}, \mathcal{X})$.

For this heterogeneous graph, the item relationships inference problem can be described as detecting the existence of substitutable and complementary links between two nodes. This link prediction problem aims to learn two kinds of node embeddings for $\mathcal{V}$, denoted as $\widehat{\mathbf{Z}}^{(s)} = \{\widehat{z}_i^{(s)} \in \mathbb{D}_k^d | i = 1, 2, \cdots, n\}$ and $\widehat{\mathbf{Z}}^{(c)} = \{\widehat{z}_i^{(c)} \in \mathbb{D}_k^b | i = 1, 2, \cdots, n\}$. Then for any two nodes v_i and v_j, the probability of existences of substitutable link and complementary link are represented by $P^{(s)}(\widehat{z}_i^{(s)}, \widehat{z}_j^{(s)})$, and $P^{(c)}(\widehat{z}_i^{(c)}, \widehat{z}_j^{(c)})$, respectively, where $P^{(s)}$ and $P^{(c)}$ are some metric function (e.g., distance) of the corresponding embedded space.

4 Methodology

In this section, we introduce DHGAN, a novel method that decouples substitutable and complementary relationships to different hyperbolic spaces and considers the mutual influence between relationships in an elaborate way at the same time. In particular, Section 4.1 gives analysis of the coupling properties between substitutable and complementary relationships in an algebraic perspective; Section 4.2 elaborates the architecture of our model in detail; Section 4.3 demonstrates our optimization and prediction procedures.

4.1 Coupling Mode in an Algebraic Perspective

To figure the nature of those binary relationships, we consider two extreme cases, i.e., perfect substitutes and perfect complements (definition in economics [22, 23]). For perfect substitutes, if we let the item itself be its own substitute, this binary relation is reflexive, symmetric and transitive, which indicates that the perfect substitutable relationship is an equivalence relation. Therefore, here exists some binary function d as a metric function over the item embedding space which reflects the strength of substitutability, and having $d(x, y) = 0$ indicates that x and y are perfect substitutes. In addition, the equivalence relation divides the entire set of items into unrelated equivalence classes. Similarly, we can assume item itself be its own complement to make the perfect complementary relationship be an equivalence relation, and the strength of relation can also be quantified by some metric function.

In real-world scenarios, perfect substitutes or perfect complements are very rare, and different users have their individualized perceptions of substitutes and complements, which means these two binary relations in an ordinary sense become 'approximate' equivalence relations. For example, substitutes usually belong to the same fine-grained category, and for some specific fine-grained categories all the items within can substitute each other. Complements usually belong to the same coarse-grained category, but items within generally cannot complement each other because their substitutes are mixed among them and we do not assume the reflexive property. However, if we approximately degenerate substitutes into a pseudo equivalence class, it is very possible that pseudo equivalence classes of the same coarse-grained category could complement each other. Therefore, we can still extract the following coupling patterns, which can be very useful in model designing:

1) Substitute neighbors of an item's substitute is its substitutes.
2) Complement neighbors of an item's substitute is its complements.
3) Substitute neighbors of an item's complement is its complements.
4) Complement neighbors of an item's complement could be its complements, or substitutes, or unrelated with this item.

4.2 Decoupled Hyperbolic Graph Attention Network

To better exploit the coupling patterns of the complicated heterogeneous item relationships, we propose a novel Decoupled Hyperbolic Graph Attention Network (DHGAN). The framework of our model is demonstrated in the left part of Figure 2.

We first decouple the item graph $\mathcal{G}$ into two hyperbolic spaces to represent substitutable relationship and complementary relationship separately:

$$\mathbf{Z}^{(s)} = f^{(s)}(\mathcal{G}),\ \mathbf{Z}^{(c)} = f^{(c)}(\mathcal{G}), \tag{1}$$

where $\mathbf{Z}^{(s)} = \{z_i^{(s)} \in \mathbb{D}_k^d | i = 1, 2, \cdots, n\}$, $\mathbf{Z}^{(c)} = \{z_i^{(c)} \in \mathbb{D}_k^b | i = 1, 2, \cdots, n\}$ are initial hyperbolic item embeddings w.r.t. two relationships, and $f^{(s)}$ and $f^{(c)}$ are exponential maps from $\mathcal{G}$ to hyperbolic spaces $\mathbb{D}_k^d$ and $\mathbb{D}_k^b$, respectively.

Then we propose two aggregation manners via two kind of designed metapaths:

- **One-Order Metapath Aggregation** in the hyperbolic space is to enhance homogeneous information within the corresponding relationship.
- **Pseudo Two-Order Metapath Aggregation** in the hyperbolic space is to propagate the mutual influence between two different relationships. The Pseudo Two-Order Metapaths are specially designed using only one-hop neighbors according to the coupling patterns. The edge between two one-hop neighbors might not exist in the real dataset, and that's why we call this metapath pseudo.

Next, we introduce the detailed implementations of the One-Order Metapath Aggregation and Pseudo Two-Order Metapath Aggregation in the hyperbolic space.

4.2.1 Hyperbolic One-Order Metapath Aggregation

To preserve more effective homogeneous information within specific relationship, we model this specific relationship using its own item subgraph that are reduced from the origin complex heterogeneous graph. Take the substitutable relationship subgraph as an example, two nodes are linked only when they can substitute each other. Similarly, two nodes in complementary relationship subgraph are linked only when they can complement each other. Then we conduct a simple one-hop neighborhood aggregation in the homogeneous subgraph, as shown in the right part of Figure 2.

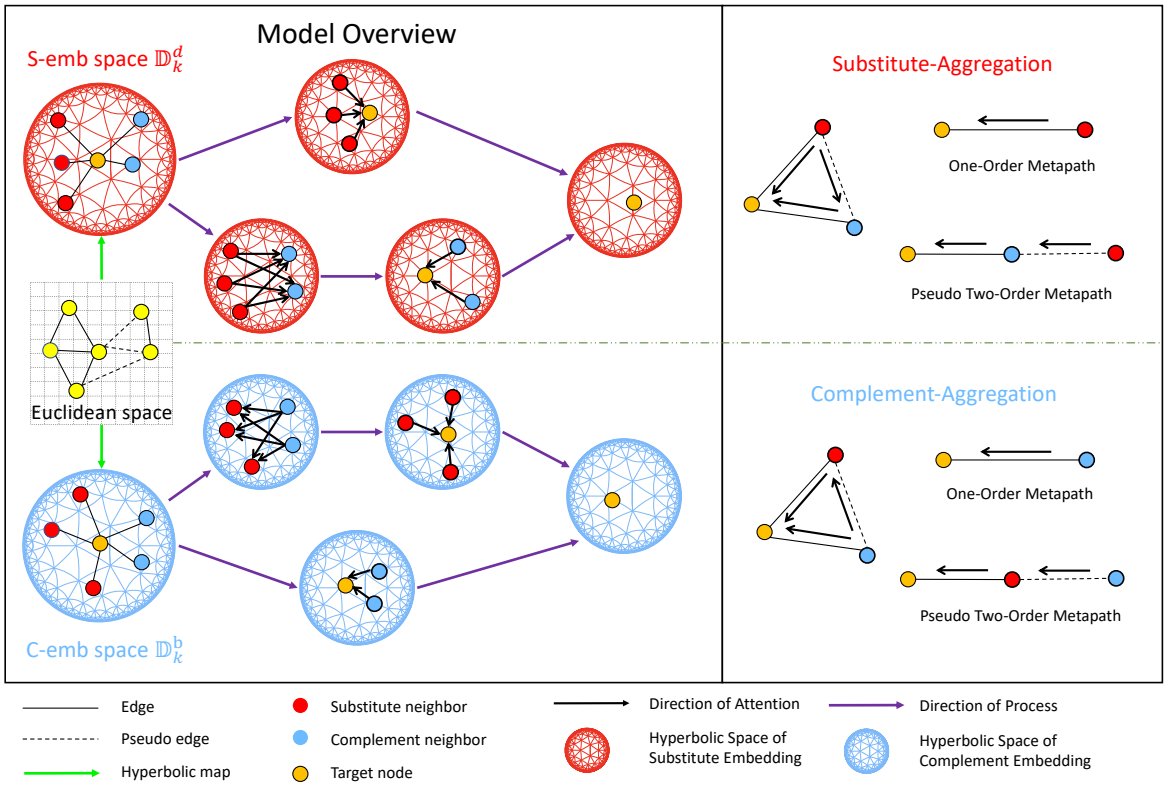


Figure 2: The architecture of DHGAN. The left part shows an overall schema of DHGAN, where item relationships graph will first be mapped to two different hyperbolic spaces, followed by One-Order Metapath aggregation layer and Pseudo Two-Order Metapath aggregation layer to capture the topological information in the same space. The right part shows the specific form of One-Order Metapath and Pseudo Two-Order Metapath.

Inspired by HAT [51], our message aggregation is designed in a hyperbolic attention mechanism. Now we take the subsititutable relationship subgraph as an example for a detailed description (similar design holds for complementary relationship subgraph). If $\mathcal{N}_m^{(s)}$ represents all substituta neighbors of a target node v_m, the hyperbolic One-Order Metapath aggregation mechanism is designed as follows:

$$\beta_{mn} = -d_k(z_m^{(s)}, z_n^{(s)}),\ f_{mn} = \frac{\exp(\beta_{mn})}{\sum_{v_{n'} \in \mathcal{N}_m^{(s)}} \exp(\beta_{mn'})},$$
$$\widetilde{z}_{m,o}^{(s)} = \sigma((\alpha \odot_k z_m^{(s)}) \oplus_k \exp_{\mathbf{0}}^k(\sum_{v_n \in \mathcal{N}_m^{(s)}} \log_{\mathbf{0}}^k(f_{mn} \odot_k z_n^{(s)}))), \tag{2}$$

where $z_m^{(s)}$ and $z_n^{(s)}$ are hyperbolic item embeddings of target node v_m and some node $v_n \in \mathcal{N}_m^{(s)}$ respectively, β_{mn} is the attention coefficient derived from the hyperbolic distance function d_k which can measure the relationship strength between two items, $\oplus_k, \odot_k$ are Möbius addition and Möbius scalar multiplication in our hyperbolic space, f_{mn} is the normalize attention coefficient, α is a learnable parameter that controls the weight of representation of target node v_m itself, σ is a nonlinear activation function ReLU, $\exp_{\mathbf{0}}^k$ and $\log_{\mathbf{0}}^k$ are exponential and logarithmic maps at the origin point respectively which connect the hyperbolic space and the tangent space, and $\widetilde{z}_{m,o}^{(s)}$ is the hyperbolic target node embedding after one-order metapath aggretion.

4.2.2 Hyperbolic Pseudo Two-Order Metapath Aggregation

To implement the coupling patterns we extract in Section 4.1, a hyperbolic Pseudo Two-Order Metapath Aggregation is conducted at the same time, as depicted in the right part of Figure 2. Unlike the usual metapaths extracted along real graph links, we construct the Pseudo Two-Order Metapath using only one-hop neighbors of the target node in the coupled heterogeneous graph. In substitutable relationship embedding space, the constructed metapath is

$$target\ node \xleftarrow{c} c\ neighbors \overset{c}{\dashleftarrow} s\ neighbors,$$

where c and s represent complementry relationship and substitutable relationship, respectively. We design this metapath under the prior knowledge of the second/third coupling pattern (the normal and dash arrows) to represent the special unusual pattern when complementary neighbors of an item's complement is its substitutes. In complementary relationship embedding space, the constructed metapath is

$$target\ node \xleftarrow{s} s\ neighbors \overset{c}{\dashleftarrow} c\ neighbors.$$

We design this metapath under the prior knowledge of the second/third coupling pattern to enhance the pattern that substitutable neighbors of an item's complement is its complements.

The message passing along the Pseudo Two-Order Metapath also follows a hyperbolic attention mechanism. We take the the substitutable embedding space as an example (similar design for complementary embedding space). For a target node v_m, we denote its substitute neighbors as $\mathcal{N}_m^{(s)}$ and its complement neighbors as $\mathcal{N}_m^{(c)}$.

First we calculate the influence from substitutable neighbors to each complementary neighbor $v_i \in \mathcal{N}_m^{(c)}$:

$$\begin{aligned} \gamma_{ij} &= -d_k(z_i^{(s)}, z_j^{(s)}),\ g_{ij} = \frac{\exp(\gamma_{ij})}{\sum_{v_{j'} \in \mathcal{N}_m^{(s)}} \exp(\gamma_{ij'})}, \\ \widetilde{z}_i^{(s)} &= \sigma((\alpha \odot z_i^{(s)}) \oplus_k \exp_{\mathbf{0}}^k(\sum_{v_j \in \mathcal{N}_m^{(s)}} \log_{\mathbf{0}}^k(g_{ij} \odot_k z_j^{(s)}))), \end{aligned} \tag{3}$$

where all the message from substitutable neighbors $z_j^{(s)} \in \mathbb{D}_k^d$ can be aggregated to each complementary neighbor and obtain the updated node embedding $\widetilde{z}_i^{(s)}$. Then we use the updated embedding $\widetilde{z}_i^{(s)}$ of the complementary neighbor to generate hyperbolic embedding of the target node v_m as follow.

$$\begin{aligned} \alpha_{mi} &= -d_k(z_m^{(s)}, \widetilde{z}_i^{(s)}),\ e_{mi} = \frac{\exp(\alpha_{mi})}{\sum_{v_{i'} \in \mathcal{N}_m^{(s)}} \exp(\alpha_{mi'})}, \\ \widetilde{z}_{m,t}^{(s)} &= \sigma((\alpha \odot z_m^{(s)}) \oplus_k \exp_{\mathbf{0}}^k(\sum_{v_i \in \mathcal{N}_m^{(s)}} \log_{\mathbf{0}}^k(e_{mi} \odot_k \widetilde{z}_i^{(s)}))). \end{aligned} \tag{4}$$

Through the hyperbolic Pseudo Two-Order Metapath aggregation, the hyperbolic target node embedding $\widetilde{z}_{m,t}^{(s)}$ contains coupling pattern information.

At last, we combine the embeddings from One-Order Metapath aggregation and Pseudo Two-Order Metapath aggregation via the following hyperbolic attention mechanism:

$$\begin{aligned} w_o^{(s)} &= (q^{(s)})^T \odot_k \tanh(W^{(s)} \otimes_k \widetilde{z}_{m,o}^{(s)}), \\ w_t^{(s)} &= (q^{(s)})^T \odot_k \tanh(W^{(s)} \otimes_k \widetilde{z}_{m,t}^{(s)}), \\ \alpha_o^{(s)} &= \frac{\exp(w_o^{(s)})}{\exp(w_o^{(s)}) + \exp(w_t^{(s)})},\ \alpha_t^{(s)} = \frac{\exp(w_t^{(s)})}{\exp(w_o^{(s)}) + \exp(w_t^{(s)})} \\ \widehat{z}_m^{(s)} &= (\alpha_o^{(s)} \odot_k \widetilde{z}_{m,o}^{(s)}) \oplus_k (\alpha_t^{(s)} \odot_k \widetilde{z}_{m,t}^{(s)}), \end{aligned} \tag{5}$$

where $W^{(s)} \in \mathbb{R}^{d \times d}$ is a trainable parameter matrix, $q^{(s)} \in \mathbb{R}^{1 \times d}$ and $b^{(s)} \in \mathbb{R}^d$ are trainable vectors, $\otimes_k$ is Möbius matrix multiplication in our hyperbolic space.

4.3 Optimization and Prediction

In our model, we use the classical cross-entropy loss and RSGD optimizer to update parameters, and apply hyperbolic distance as the metric to do link prediction.

To obtain the optimal representations $\widehat{z}_i^s$ and $\widehat{z}_i^c$ of the target node v_i, we sample its one-hop neighbors $v_j \in \mathcal{N}_i$ as positive samples, and randomly sample its negative nodes from complementary one-hop neighbors, i.e., $v_l \notin \mathcal{N}_i^{(s)}$. Then our cross-entropy loss is designed as follows:

$$\begin{aligned} \mathcal{L}^{(s)} &= \sum_{j,l} -\log[1 - \sigma(d_k(\widehat{z}_i^{(s)}, \widehat{z}_j^{(s)})] - \log[\sigma(d_k(\widehat{z}_i^{(s)}, \widehat{z}_l^{(s)}) - 0.5], \\ \mathcal{L}^{(c)} &= \sum_{j,l} -\log[1 - \sigma(d_k(\widehat{z}_i^{(c)}, \widehat{z}_j^{(c)})] - \log[\sigma(d_k(\widehat{z}_i^{(c)}, \widehat{z}_l^{(c)}) - 0.5], \\ \mathcal{L} &= \mathcal{L}^{(s)} + \mathcal{L}^{(c)}. \end{aligned}$$

Then we use Riemannian SGD (RSGD) [3, 27, 44] to update model parameters. RSGD mimics the stochastic gradient descent optimization on the geometry of hyperbolic manifold. After training, we can use the learned hyperbolic embeddings to predict relationship between items. For any two items v_i and v_j, the prediction scores of substitutability and complementarity between them can be calculated by the hyperbolic distance over corresponding relationship spaces.

Datasets	Electronics	Sports.	Toys.
#Items	260144	390953	238347
#Train Substitute edges	2266944	2146738	1263930
#Train Complement edges	845328	2542888	1645048
#Test Substitute edges	71866	78520	59440
#Test Complement edges	25938	51150	48256
#Category_id2	22	3	20
#Category_id3	155	98	384
#Brand_id	26799	39958	25518

Table 1: Statistics of datasets

5 Experiments

In this section, we first give the experimental setup, then evaluate the effectiveness of the proposed approach, DHGAN, compared with the state-of-the-art graph embedding methods. Next, we give the ablation study and hyper-parameter analysis. We finally give the robustness analysis in the scenario of missing or reversed item relationships.

5.1 Experimental Setup

Datasets. We conduct our experiments on three public datasets collected from Amazon: Electronics, Sports_and_Outdoors and Toys_and_Games, which have varing sizes and characteristics. These datasets are available online [1] and have been widely used in many existing studies [19, 24, 50]. We follow previous works [19, 24] to regard substitutable and complementary relationships as the 'also-viewed' and 'also-bought' relationships, respectively. For each item, we use its categories and brand as features. The statistical information of these datasets is summarized in Table 1.

Baselines. For sufficient comparison, We compare DHGAN with state-of-the-art graph representation learning methods described as follows. GraphSage, GAT, HAN, GATNE-I, DecGCN are Euclidean-based embedding methods, while HGNN, HAT are Hyperbolic-based embedding methods.

[1] https://nijianmo.github.io/amazon/index.html

- **GraphSage** [12]. GraphSAGE provides a general inductive approach to combine node features and structural information. It generates node embeddings by sampling and aggregating feature information from the local neighborhood. To obtain corresponding node embeddings, we apply GraphSAGE on two homogeneous graphs induced from heterogeneous item graph with respect to each relationship.
- **GAT** [39]. GAT is a novel convolution-style neural network designed for the homogeneous graphs. It computes the hidden representation of each node by leveraging masked self-attention over its neighbors. We also apply GAT on two homogeneous graphs to get nodes embedding for each relationship.
- **HAN** [40]. HAN is a representative heterogeneous graph neural network based solely on attention mechanism. It leverages node-level attention and semantic-level attention to learn the importance of metapath based nodes and metapaths themselves, respectively.
- **GATNE-I** [4]. GATNE with inductive settings (GATNE-I) is a state-of-the-art method to model attributed multiplex heterogeneous network. For each node, despite sharing attribute embedding and structural base embedding, it also leverages separate GCNs [15] to learn multiple edge embeddings w.r.t. different types of relations.
- **DecGCN** [19]. DecGCN generates item embeddings in separated embedding spaces w.r.t. different item relationships. It captures the mutual inference between both graph structures and different item semantics.
- **HGNN** [18]. HGNN is a homogeneous hyperbolic-based graph convolutional network, which performs feature aggregation and transformation in hyperbolic space. HGNN can well capture hierarchical with power law distribution which exists naturally in graph data.
- **HAT** [51]. HAT also conducts both feature aggregation and transformation in hyperbolic space, and provides a hyperbolic proximity based attention mechanism to aggregate node features.
- **DHGAN-Euclidean**. DHGAN-Euclidean is a variant of our model that performs same feature transformation and aggregation in Euclidean space. It also replaces the hyperbolic distance in attention mechanism with negative inner product, and replaces Riemann-SGD with Adam optimizer.

Parameter Configuration. For fair comparison, the hidden node embedding dimension of all the models is set to 64 and the output embedding dimension is set to 32. The curvatures in DHGAN and other hyperbolic-based methods (i.e., HGNN and HAT) are all set to -1. For these GCN-based methods, we stack two convolutional layers, each of which aggregates 10 sampled neighbors for each type of edge. For the model optimization, all models will be trained 2 epochs with a learning rate of 0.01 and a batch size of 1024. Euclidean-based graph baselines use Adam optimizer, our model and Hyperbolic-based graph baselines use Riemann-SGD optimizer. In the experiments, all models are conducted with DGL platform [2] and PyTorch framework [3].

[2]https://docs.dgl.ai/
[3]https://pytorch.org/tutorials/

Evaluation Metrics. We conduct the link prediction task to validate the performance of all models. For each item, we randomly sample one edge for each type of relationships to construct the test data, and use the rest for the training data. During the training stage, positive sample consists of one source node and one true destination node, and the corresponding negative samples consist of the same source node and 5 randomly sampled destination nodes. While testing, for each item, we fuse its ground truth with 1000 sampled negative instances [19, 24], and use HitRate@K, MRR@K and NDCG@K as evaluation metrics.

5.2 Overall Performance

The experimental results for substitutable and complementary item relationships prediction are shown in Table 2, where all the results are averaged across 5 repeat runs. The best result of each evaluation metric for each dataset is highlighted in **boldface**, and the best performance among all baselines is highlighted with underlines.

As we can see, DHGAN significantly outperforms the best baselines on Toys_and_Games and Sports_and_Outdoors datasets. Among all metrics, only HR@10 on Electronics dataset when inferring complementary items is slightly smaller than HAN. Take the task of inferring substitutable items as an example, on average, DHGAN achieves better performance than the best results among all the baselines by 5.4%, 17.8% and 15.2% in terms of HR@10, MRR@10 and NDCG@10 respectively on all datasets, which illustrates the superiority of our proposed model. Especially, our model improves significantly on MRR@10 and NDCG@10 in these three datasets, indicating that DHGAN can not only distinguish substitutable and complementary items, but also rank the correct samples in top positions.

On one hand, compared with Euclidean-based baselines (i.e., GraphSage, GAT, HAN, GATNE-I, DecGCN), on average, DHGAN achieves 16.89%, 31.38%, 28.18% gain in HR@10, MRR@10, NDCG@10, respectively in inferring substitutable items, and 8.93%, 24.82%, 21.04% gain in inferring complementary items. Among Euclidean-based baselines, HAN, GATNE, DecGCN perform relatively better than GraphSage and GAT since these methods consider the heterogeneity of item relationships and model the interactions between different relationships through pre-defined metapaths or leveraging node-level and semantic-level attention mechanism. However, these methods ignore the prior knowledge of how relationships affect each other, and all these Euclidean-based baselines can not capture the hierarchical with power law distribution that exists in graph data naturally, resulting that these methods do not have sufficient expressive power to model heterogeneous item relationships. This is why DHGAN outperforms all these Euclidean-based methods.

On the other hand, DHGAN outperforms hyperbolic-based baselines, such as HGNN and HAT, by a large margin. Although they map node embedding into the hyperbolic space and can model the natural hierarchical graph structure, they neglect the mutual influence between different item relationships, resulting in relatively poor performances.

5.3 Ablation Study

To clearly illustrate the effectiveness of the hyperbolic embedding mechanism and our designed Pseudo Two-Order Metapath

		Substitute			Complement		
dataset	model	HR@10	MRR@10	NDCG@10	HR@10	MRR@10	NDCG@10
Electronics	GraphSage	0.664	0.341	0.417	0.693	0.361	0.439
	GAT	0.703	0.369	0.448	0.725	0.364	0.449
	HAN	0.761	0.415	0.497	**0.785**	0.437	0.520
	GATNE-I	0.561	0.248	0.322	0.769	0.363	0.459
	DecGCN	0.745	0.449	0.519	0.667	0.365	0.432
	HGNN	0.726	0.423	0.495	0.604	0.320	0.387
	HAT	0.796	0.510	0.578	0.609	0.322	0.390
	DHGAN-Euclidean	0.829	0.620	0.671	0.748	0.502	0.561
	DHGAN	**0.855**	**0.668**	**0.713**	0.780	**0.590**	**0.635**
Sports_and_Outdoors	GraphSage	0.697	0.383	0.457	0.630	0.296	0.374
	GAT	0.401	0.256	0.291	0.679	0.323	0.407
	HAN	0.813	0.464	0.547	0.771	0.371	0.465
	GATNE-I	0.632	0.287	0.368	0.805	0.441	0.528
	DecGCN	0.738	0.416	0.480	0.712	0.385	0.461
	HGNN	0.690	0.426	0.489	0.648	0.361	0.429
	HAT	0.745	0.480	0.543	0.679	0.375	0.447
	DHGAN-Euclidean	0.861	0.680	0.724	0.745	0.494	0.551
	DHGAN	**0.866**	**0.706**	**0.745**	**0.847**	**0.654**	**0.700**
Toys_and_Games	GraphSage	0.777	0.442	0.522	0.780	0.454	0.532
	GAT	0.818	0.477	0.558	0.790	0.461	0.540
	HAN	0.859	0.534	0.612	0.798	0.448	0.531
	GATNE-I	0.690	0.350	0.431	0.799	0.406	0.499
	DecGCN	0.784	0.491	0.563	0.802	0.512	0.579
	HGNN	0.800	0.501	0.578	0.769	0.477	0.547
	HAT	0.864	0.604	0.667	0.827	0.557	0.621
	DHGAN-Euclidean	0.903	0.727	0.769	0.826	0.597	0.652
	DHGAN	**0.914**	**0.756**	**0.794**	**0.882**	**0.698**	**0.742**

Table 2: Results of item relationship prediction in terms of HR@10, MRR@10 and NDCG@10, where negative samples = 1000.

aggregation, we conduct two ablation experiments using DHGAN-Euclidean and the degenerated DHGAN without the Pseudo Two-Order Metapath aggregation (i.e., HAT), respectively. The results are also shown in Table 2. As we can see, DHGAN outperforms DHGAN-Euclidean over three datasets among all evaluation metrics, especially when inferring complementary items. The reason is that complementary relationship is not transitive, i.e. more complicated than substitutable relationship, which requires a stronger model expressive capacity to capture item representation delicately. In this respect, models based on hyperbolic spaces outperform those based on Euclidean spaces. Our DHGAN also outperforms HAT over three datasets among all evaluation metrics, which indicate that our Pseudo Two-Order Metapath aggregation helps obtaining effective item embedding by introducing prior decoupling knowledge. Furthermore, the fact that DHGAN-Euclidean outperforms all the baselines among most metrics in three datasets, also shows the effectiveness of our Pseudo Two-Order Metapath aggregation design.

5.4 Hyperparameter Analysis

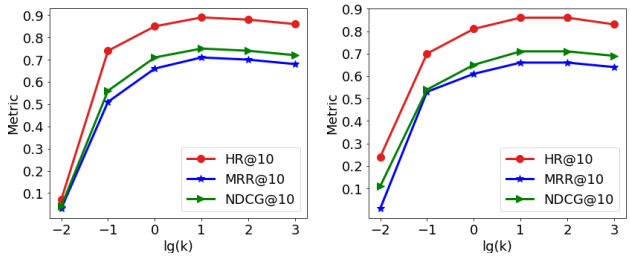
To evaluate the sensitivity of our model DHGAN, we further conduct experiments while vary the key hyperparameters of DHGAN. First, we investigate the impact of the neighborhood size in each convolutional layer. We train DHGAN with different number of sampled neighbors, denoted as {5, 5}, {5, 10}, {10, 5} and {10, 10} (default). For example, {10, 5} means that DHGAN aggregate 10 one-hop neighbors and 5 two-hop neighbors. The results is shown in Table 3. As is shown, two-layer DHGAN with {5, 5} performs worst on these three datasets, since it does not aggregate enough information from local neighbors. DHGAN with {10, 10} performs best in most scenarios (except for inferring complementary items on Electronics dataset), which indicates that aggregating more one-hop and two-hop neighbors can provide more effective information for the modeling. In addition, in order to analyze the impact of the curvature, we vary its value in {-100, -10, -1, -1/10, -1/100, -1/1000}. We take Electronics dataset as an example (other two datasets show similar results). The results are shown in Figure 3, where k is the negative reciprocal of the curvature. As is described, when k is very small (i.e., 0.01), the performance is very pool. When k is large (i.e., 1000), the performance also begins to drop, which means proper curvature matters for our hyperbolic model. If the embedding space is more Euclidean, the expressive capacity of model will be weaken. If the embedding space is more strongly hyperbolic and the curvature far exceed the extent of the degree distribution of items, the model also will not perform well.

5.5 Robustness Analysis

In some real scenarios, heterogeneous item relationships are noisy, which may negatively influence the model. For example, many substitutable or complementary relationships may be missing or reversed. To further validate the robustness of the proposed model, we first randomly remove 10%-80% substitutable and complementary

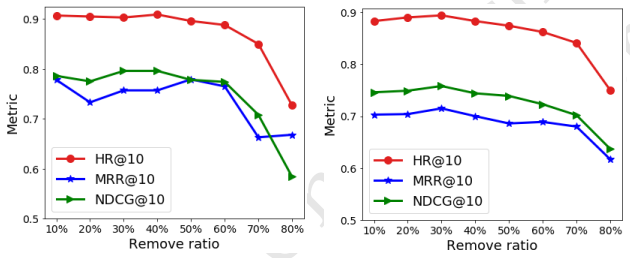
		Substitute			Complement		
dataset	neighbors	HR@10	MRR@10	NDCG@10	HR@10	MRR@10	NDCG@10
Electronics	{5, 5}	0.809	0.571	0.628	0.770	0.564	0.613
	{5, 10}	0.843	0.620	0.667	**0.815**	**0.620**	**0.667**
	{10, 5}	0.816	0.587	0.642	0.777	0.584	0.630
	{10, 10}	**0.855**	**0.668**	**0.713**	0.780	0.590	0.635
Sports_and_Outdoors	{5, 5}	0.841	0.652	0.697	0.719	0.501	0.553
	{5, 10}	0.863	0.696	0.736	0.811	0.597	0.648
	{10, 5}	0.856	0.674	0.717	0.805	0.604	0.652
	{10, 10}	**0.866**	**0.706**	**0.745**	**0.847**	**0.654**	**0.700**
Toys_and_Games	{5, 5}	0.863	0.663	0.711	0.831	0.621	0.671
	{5, 10}	0.848	0.644	0.693	0.815	0.620	0.667
	{10, 5}	0.858	0.678	0.722	0.836	0.630	0.680
	{10, 10}	**0.914**	**0.756**	**0.794**	**0.882**	**0.698**	**0.742**

Table 3: Results of DHGAN when varing the number of sampled neighbors.



(a) Substitute (E) (b) Complement (E)

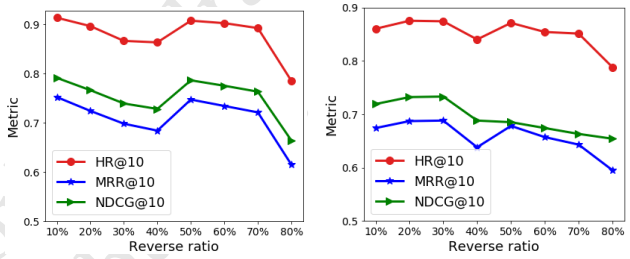
Figure 3: The performance of DHGAN when varing the curvature.



(a) Substitute (T) (b) Complement (T)

Figure 4: The performance of DHGAN when removing substitutable and complementary relationships.

relationships in the Toys_and_Games dataset as an example. The results are shown in Figure 4. We can see that, with the remove ratio increases from 10% to 60%, the performance of DHGAN keeps stable and satisfactory. As the ratio continues to increase, the performance decreases since the model can not learn correct relationships. The superior and stable performance comes from two aspects. First, we sample neighbors of fixed-size in each convolutional layer, which helps to reduce the influence of noise. Second, our decoupling design can make full use of the available training data to capture latent interacted patterns between item relationships. We also reverse substitutable and complementary relationships with the reverse ratio from 10% to 80% in Toys_and_Games dataset. For example, when the reverse ratio is 10%, we turn 10% substitutable relationships



(a) Substitute (T) (b) Complement (T)

Figure 5: The performance of DHGAN when reversing substitutable and complementary relationships.

into complementary relationships and vice versa. The results are in Figure 5. As is shown, the performance trend is similar to that of removing relationships. It is worth noting that the fluctuation may be caused by intrinsic noise in data when the reverse ratio is about 40%. Overall, the experiments validate the robustness and effectiveness of the proposed model.

6 Conclusion

In this paper, we propose a novel Decoupled Hyperbolic Graph Attention Network (DHGAN) for the task of inferring heterogeneous relationships between items. DHGAN introduces rational prior knowledge and designs Pseudo Two-Order Metapath aggregation from an algebraic perspective, which achieves better model interpretability when decoupling the heterogeneous relationships. DHGAN also introduces different hyperbolic spaces for different types of relationship, which can capture the natural hierarchical structure of graph data. Extensive experiments on three public datasets of Amazon demonstrate the remarkable performance of our method over existing state-of-the-art Euclidean graph methods and other hyperbolic-based methods, and validate the robustness of the proposed model.

References

[1] Muhammad Asaduzzaman, Simon Catterall, Jay Hubisz, Roice Nelson, and Judah Unmuth-Yockey. 2020. Holography on tessellations of hyperbolic space. *Physical Review D* 102, 3 (2020), 034511.
[2] Albert-László Barabási and Eric Bonabeau. 2003. Scale-free networks. *Scientific american* 288, 5 (2003), 60–69.

[3] Silvere Bonnabel. 2013. Stochastic gradient descent on Riemannian manifolds. *IEEE Trans. Automat. Control* 58, 9 (2013), 2217–2229.

[4] Yukuo Cen, Xu Zou, Jianwei Zhang, Hongxia Yang, Jingren Zhou, and Jie Tang. 2019. Representation learning for attributed multiplex heterogeneous network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1358–1368.

[5] Benjamin Paul Chamberlain, Stephen R Hardwick, David R Wardrope, Fabon Dzogang, Fabio Daolio, and Saúl Vargas. 2019. Scalable hyperbolic recommender systems. *arXiv preprint arXiv:1902.08648* (2019).

[6] Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. 2019. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems* 32 (2019), 4868–4879.

[7] Bhuwan Dhingra, Christopher J Shallue, Mohammad Norouzi, Andrew M Dai, and George E Dahl. 2018. Embedding text in hyperbolic spaces. *arXiv preprint arXiv:1806.04313* (2018).

[8] Yuxiao Dong, Nitesh V Chawla, and Ananthram Swami. 2017. metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. 135–144.

[9] Alessandro Epasto and Bryan Perozzi. 2019. Is a single embedding enough? learning node representations that capture multiple social contexts. In *The World Wide Web Conference*. 394–404.

[10] Shanshan Feng, Lucas Vinh Tran, Gao Cong, Lisi Chen, Jing Li, and Fan Li. 2020. Hme: A hyperbolic metric embedding approach for next-poi recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1429–1438.

[11] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. 855–864.

[12] William L Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *arXiv preprint arXiv:1706.02216* (2017).

[13] Wang-Cheng Kang, Eric Kim, Jure Leskovec, Charles Rosenberg, and Julian McAuley. 2019. Complete the look: Scene-based complementary product recommendation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10532–10541.

[14] Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. 2020. Hyperbolic image embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6418–6428.

[15] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[16] Dmitri Krioukov, Fragkiskos Papadopoulos, Maksim Kitsak, Amin Vahdat, and Marián Boguná. 2010. Hyperbolic geometry of complex networks. *Physical Review E* 82, 3 (2010), 036106.

[17] Ninghao Liu, Qiaoyu Tan, Yuening Li, Hongxia Yang, Jingren Zhou, and Xia Hu. 2019. Is a single vector enough? exploring node polysemy for network embedding. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 932–940.

[18] Qi Liu, Maximilian Nickel, and Douwe Kiela. 2019. Hyperbolic graph neural networks. In *Advances in Neural Information Processing Systems*. 8228–8239.

[19] Yiding Liu, Yulong Gu, Zhuoye Ding, Junchao Gao, Ziyi Guo, Yongjun Bao, and Weipeng Yan. 2020. Decoupled Graph Convolution Network for Inferring Substitutable and Complementary Items. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2621–2628.

[20] Jianxin Ma, Peng Cui, Kun Kuang, Xin Wang, and Wenwu Zhu. 2019. Disentangled graph convolutional networks. In *International Conference on Machine Learning*. PMLR, 4212–4221.

[21] Yao Ma, Zhaochun Ren, Ziheng Jiang, Jiliang Tang, and Dawei Yin. 2018. Multi-dimensional network embedding with hierarchical structure. In *Proceedings of the eleventh ACM international conference on web search and data mining*. 387–395.

[22] N Gregory Mankiw. 2014. *Principles of economics*. Cengage Learning.

[23] Andreu Mas-Colell, Michael Dennis Whinston, Jerry R Green, et al. 1995. *Microeconomic theory*. Vol. 1. Oxford university press New York.

[24] Julian McAuley, Rahul Pandey, and Jure Leskovec. 2015. Inferring networks of substitutable and complementary products. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 785–794.

[25] Leyla Mirvakhabova, Evgeny Frolov, Valentin Khrulkov, Ivan Oseledets, and Alexander Tuzhilin. 2020. Performance of hyperbolic geometry models on top-N recommendation tasks. In *Fourteenth ACM Conference on Recommender Systems*. 527–532.

[26] Maximillian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. *Advances in neural information processing systems* 30 (2017), 6338–6347.

[27] Maximillian Nickel and Douwe Kiela. 2018. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *International Conference on Machine Learning*. PMLR, 3779–3788.

[28] Fragkiskos Papadopoulos, Maksim Kitsak, M Ángeles Serrano, Marián Boguná, and Dmitri Krioukov. 2012. Popularity versus similarity in growing networks. *Nature* 489, 7417 (2012), 537–540.

[29] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 701–710.

[30] Peter Petersen. 2006. *Riemannian geometry*. Vol. 171. Springer.

[31] Vineeth Rakesh, Suhang Wang, Kai Shu, and Huan Liu. 2019. Linked variational autoencoders for inferring substitutable and supplementary items. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. 438–446.

[32] Erzsébet Ravasz and Albert-László Barabási. 2003. Hierarchical organization in complex networks. *Physical review E* 67, 2 (2003), 026112.

[33] Jianing Sun, Zhaoyue Cheng, Saba Zuberi, Felipe Pérez, and Maksims Volkovs. 2021. HGCF: Hyperbolic Graph Convolution Networks for Collaborative Filtering. In *Proceedings of the Web Conference 2021*. 593–601.

[34] Zequn Sun, Muhao Chen, Wei Hu, Chengming Wang, Jian Dai, and Wei Zhang. 2020. Knowledge association with hyperbolic knowledge graph embeddings. *arXiv preprint arXiv:2010.02162* (2020).

[35] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*. 1067–1077.

[36] Abraham A Ungar. 2001. Hyperbolic trigonometry and its application in the Poincaré ball model of hyperbolic geometry. *Computers & Mathematics with Applications* 41, 1-2 (2001), 135–147.

[37] Abraham Albert Ungar. 2008. *Analytic hyperbolic geometry and Albert Einstein's special theory of relativity*. World scientific.

[38] Abraham Albert Ungar. 2008. A gyrovector space approach to hyperbolic geometry. *Synthesis Lectures on Mathematics and Statistics* 1, 1 (2008), 1–194.

[39] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).

[40] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. 2019. Heterogeneous graph attention network. In *The World Wide Web Conference*. 2022–2032.

[41] Xiang Wang, Hongye Jin, An Zhang, Xiangnan He, Tong Xu, and Tat-Seng Chua. 2020. Disentangled graph collaborative filtering. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1001–1010.

[42] Xiao Wang, Yiding Zhang, and Chuan Shi. 2019. Hyperbolic heterogeneous information network embedding. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 5337–5344.

[43] Zihan Wang, Ziheng Jiang, Zhaochun Ren, Jiliang Tang, and Dawei Yin. 2018. A path-constrained framework for discriminating substitutable and complementary products in e-commerce. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. 619–627.

[44] Benjamin Wilson and Matthias Leimeister. 2018. Gradient descent in hyperbolic space. *arXiv preprint arXiv:1805.08207* (2018).

[45] Chen Wu and Ming Yan. 2017. Session-aware information embedding for e-commerce product recommendation. In *Proceedings of the 2017 ACM on conference on information and knowledge management*. 2379–2382.

[46] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. 2020. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems* 32, 1 (2020), 4–24.

[47] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 974–983.

[48] Mingyue Zhang, Xuan Wei, Xunhua Guo, Guoqing Chen, and Qiang Wei. 2019. Identifying complements and substitutes of products: A neural network framework based on product embedding. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 13, 3 (2019), 1–29.

[49] Shijie Zhang, Hongzhi Yin, Qinyong Wang, Tong Chen, Hongxu Chen, and Quoc Viet Hung Nguyen. 2019. Inferring Substitutable Products with Deep Network Embedding.. In *IJCAI*. 4306–4312.

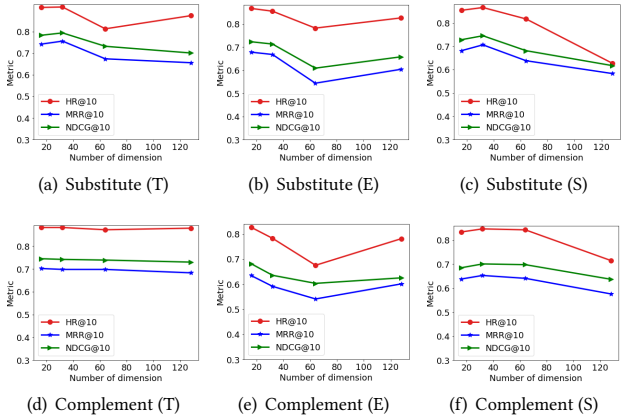
[50] Yin Zhang, Haokai Lu, Wei Niu, and James Caverlee. 2018. Quality-aware neural complementary item recommendation. In *Proceedings of the 12th ACM Conference on Recommender Systems*. 77–85.

[51] Yiding Zhang, Xiao Wang, Chuan Shi, Xunqiang Jiang, and Yanfang Fanny Ye. 2021. Hyperbolic graph attention network. *IEEE Transactions on Big Data* (2021).

[52] Tong Zhao, Julian McAuley, Mengya Li, and Irwin King. 2017. Improving recommendation accuracy using networks of substitutable and complementary products. In *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 3649–3655.

[53] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2020. Graph neural networks: A review of methods and applications. *AI Open* 1 (2020), 57–81.

A Additional Experiments



(a) Substitute (T) (b) Substitute (E) (c) Substitute (S)

(d) Complement (T) (e) Complement (E) (f) Complement (S)

Figure 6: The performance of DHGAN when varing the item embedding dimension

B Notations

We also investigate the impact of latent item embedding dimension. We vary the number of embedding dimension from 16 to 128. The results are shown in Figure 6, where E, S, T represent Electronics, Sports_and_Outdoors, Toys_and_Games, respectively. As shown, DHGAN perform best when the dimension of item embedding is close to 32. We can see that the performance of DHGAN decreases or fluctuates with the increase of embedding dimension. Therefore, we set the final item embedding dimension of all datasets to 32 to balance efficiency and effectiveness.

C RSGD

Taking the substitute embedding at iteration t, i.e. $Z_t^{(s)}$ as an example, RSGD consists of the following steps:

1) Compute the gradient of loss function in Euclidean manner:

$$\nabla L = \frac{\partial L}{\partial Z_t^{(s)}} \in \mathbb{R}^d.$$

2) Map the Euclidean gradient into the tangent space $\mathbb{D}_k^d$, and obtain the Riemannian gradient:

$$\nabla^{\mathbb{D}_k^d} L = \frac{(1 - ||Z_t^{(s)}||^2)^2}{4} \nabla L.$$

3) Estimate the update step in hyperbolic space using the exponential map and update the embedding with step size η:

$$Z_{t+1}^{(s)} = \exp_{Z_t^{(s)}}(-\eta \nabla^{\mathbb{D}_k^d} L).$$

notation	description
$(\mathcal{M}, g)$	a Riemannian manifold
$\mathcal{M}$	a real, smooth manifold
g	a Riemannian metric
$\mathcal{T}_x\mathcal{M}$	the tangent space at point x
g_x	a positive-definite inner product on the tangent space at point x
q	curvature
k	$k = -\frac{1}{q}$
$(\mathbb{D}_k^d, g^k)$	the Gyrovector space
$\mathbb{D}_k^d$	$\mathbb{D}_k^d = \{x \in \mathbb{R}^d : k\|x\| < 1, k \geqslant 0\}$
$\|\|\|$	the Euclidean norm
λ_x	$\lambda_x := \frac{2}{1-k\|x\|^2}$ at point $x \in \mathbb{D}_k^d$
g_x^k	$g_x^k = \lambda_x^2 g^E$
g^E	$g^E = \mathbf{I}_d$ the Euclidean metric tensor
$\oplus_k$	the Möbius addition
$\otimes_k$	the Möbius scalar or Matrix multiplication
d_k	the hyperbolic distance function
$\exp_x^k$	the exponential map from the tangent space at point x to hyperbolic space
tanh	the hyperbolic tangent function
$\log_x^k$	the logarithmic map from hyperbolic space to its tangent space at point x
$\tanh^{-1}$	the inverse hyperbolic tangent function
$\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{X}, \psi)$	a heterogeneous graph generated by multiple item relationships
$\mathcal{V}$	the set of n items
$\mathcal{E}$	the set of edges
$\mathcal{X}$	the set of nodes' attributes
$\mathcal{R}$	the set of relationships
ψ	a function that map each edge into the set of relationships
$\mathcal{G}^{(s)}$	the substitutable item subgraph
$\mathcal{G}^{(c)}$	the complementary item subgraph
$\mathcal{E}^{(s)}$	the set of substitutable edge
$\mathcal{E}^{(c)}$	the set of complementary edge
$P^{(s)}$	metric function (e.g., distance) of substitutable embedding space
$P^{(c)}$	metric function (e.g., distance) of complementary embedding space
$f^{(s)}$	an exponential maps from $\mathcal{G}$ to $\mathbb{D}_k^d$
$f^{(c)}$	an exponential maps from $\mathcal{G}$ to $\mathbb{D}_k^b$
$\widehat{\mathbf{Z}}^{(s)}$	substitutable embeddings for $\mathcal{V}$
$\widehat{\mathbf{Z}}^{(c)}$	complementary embeddings for $\mathcal{V}$
$z_m^{(s)}, \widehat{z}_m^{(s)}$	substitutable embeddings of node v_m
$z_m^{(c)}, \widehat{z}_m^{(c)}$	complementary embeddings of node v_m
$\widetilde{z}_{m,o}^{(s)}$	substitutable embeddings of node v_m in One-Order Metapath
$\widetilde{z}_{m,t}^{(s)}$	substitutable embeddings of node v_m in Pseudo Two-Order Metapath
v_m	a item m in $\mathcal{V}$
$\mathcal{N}_m^{(s)}$	all substitute neighbors of a node v_m
$\mathcal{N}_m^{(c)}$	all complement neighbors of a node v_m
σ	a nonlinear activation function ReLU
α	a learnable parameter in $\mathbb{R}^d$
$\alpha_{ij}, \beta_{ij}, \gamma_{ij}$	the attention coefficient of node v_j to node v_i
$\mathcal{L}^{(s)}$	the substitutability loss function
$\mathcal{L}^{(c)}$	the complementarity loss function
$\mathcal{L}$	the total loss function
∇L	gradient operator
$\frac{\partial L}{\partial t}$	The partial derivative of the L function with respect to t
η	learning step size

Table 4: Notations