Complex Hyperbolic Knowledge Graph Embeddings with Fast Fourier Transform

Huiru Xiao and **Xin Liu** and **Yangqiu Song**
Hong Kong University of Science and Technology, Hong Kong SAR
{hxiaoaf,xliucr,yqsong}@cse.ust.hk

Ginny Y. Wong and **Simon See**
NVIDIA AI Technology Center (NVATIC), NVIDIA, Santa Clara, USA
{gwong,ssee}@nvidia.com

Abstract

The choice of geometric space for knowledge graph (KG) embeddings can have significant effects on the performance of KG completion tasks. The hyperbolic geometry has been shown to capture the hierarchical patterns due to its tree-like metrics, which addressed the limitations of the Euclidean embedding models. Recent explorations of the complex hyperbolic geometry further improved the hyperbolic embeddings for capturing a variety of hierarchical structures. However, the performance of the hyperbolic KG embedding models for non-transitive relations is still unpromising, while the complex hyperbolic embeddings do not deal with multi-relations. This paper aims to utilize the representation capacity of the complex hyperbolic geometry in multi-relational KG embeddings. To apply the geometric transformations which account for different relations and the attention mechanism in the complex hyperbolic space, we propose to use the fast Fourier transform (FFT) as the conversion between the real and complex hyperbolic space. Constructing the attention-based transformations in the complex space is very challenging, while the proposed Fourier transform-based complex hyperbolic approaches provide a simple and effective solution. Experimental results show that our methods outperform the baselines, including the Euclidean and the real hyperbolic embedding models.

1 Introduction

Knowledge graph (KG) representation learning is important to the KG inference as well as the downstream tasks (Nickel et al., 2016). It has been noticed that the embedding space has significant effects on the performance of KG completion tasks. Previous works have proposed the KG embedding models in Euclidean space (Bordes et al., 2013; Nickel et al., 2011; Yang et al., 2015), complex Euclidean space (Trouillon et al., 2016; Sun et al., 2019), hyperbolic space (Balazevic et al., 2019; Chami et al., 2020).[1] These models learn the embeddings of the KG entities in the selected geometric spaces and parameterize the relation representations as the geometric transformations, such as translation, rotation, matrix multiplication, etc.

Embedding space	Hierarchical patterns	Multi-relation properties
Euclidean	🤔 Too flat and narrow to represent hierarchical patterns.	★ Capture some relation properties.
Complex Euclidean	🤔 Too flat and narrow to represent hierarchical patterns.	★★ Capture more relation properties.
Hyperbolic	★★ Improve the Euclidean embeddings a lot; Limitations on hierarchies that deviate from tree metrics.	★★ Capture transitivity well; Limitations on other relation properties.
Complex hyperbolic	★★★ Further improve the hyperbolic embeddings to handle various and flexible hierarchical structures.	🤔 Capture transitivity very well; Not applicable to multi-relations.
Complex hyperbolic + Fourier transform	★★★ Maintain the representation capacity for various hierarchical structures.	★★★ Extend to multi-relations and achieve good results.

Figure 1: The summary of embedding spaces for hierarchical patterns and multi-relation properties.

The Euclidean and complex Euclidean embedding models can capture relation properties including symmetry/anti-symmetry, inversion, and composition, but they cannot handle the transitive relations such as hypernymy. Generally, the transitive relation forms a tree-like structure, for which hyperbolic geometry has a more powerful representation capacity than Euclidean geometry because the hyperbolic space can be regarded as a continuous approximation to trees (Krioukov et al., 2010).

However, most real-world graphs with transitivity do not necessarily form exact tree structures since the transitive relations can lead to a globally hierarchical structure with varying local structures, such as multitree structures (Griggs et al., 2012) and taxonomies (Suchanek et al., 2007). Thus, the

[1] To avoid wordiness, in this paper, we use *hyperbolic space* to refer to *real hyperbolic space*, *hyperbolic geometry* to refer to *real hyperbolic geometry*, and *hyperbolic embeddings* to refer to *real hyperbolic embeddings*.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 5228 - 5239
December 7-11, 2022

hyperbolic geometry which resembles tree metrics still has limitations on capturing various and flexible hierarchical structures. To tackle the limitation of hyperbolic embeddings, a recent work (Xiao et al., 2021) proposed to explore the complex hyperbolic geometry to learn the embeddings of hierarchical graphs. Due to the variable negative curvature (Goldman, 1999), the complex hyperbolic space is more flexible in handling varying structures while the tree-like properties are still retained. Despite the remarkable improvements in single transitive relation inference, the complex hyperbolic geometry has not been utilized for multi-relational embeddings.

In this paper, we are motivated to make use of the complex hyperbolic geometry's representation superiority in KGs. There are two main challenges in extending the complex hyperbolic embeddings to multiple relations. First, the geometric transformations in complex hyperbolic geometry are complicated and challenging to optimize due to the numerical instabilities, making it difficult to apply the complex geometric transformations for different relations. Second, it is hard to build the neural network unit or layer in the complex domain. Missing the complex attention mechanism would restrict the parameterization capability and make the complex domain-based model difficult to generalize to further downstream tasks.

To address the above problems, we propose a complex hyperbolic KG embedding approach with the fast Fourier transform. Our approach can utilize the representation capacity of the complex hyperbolic geometry as well as the well-developed attention-based geometric transformations as relation parameterization, while we borrow the fast Fourier transform (FFT) and inverse fast Fourier transform (IFFT) to provide the conversion between the real and complex hyperbolic space. We regard the complex hyperbolic embeddings in the unit ball model (a projective geometry-based model to identify the complex hyperbolic space) (Goldman, 1999) and the hyperbolic embeddings in the Poincaré ball model (a model of the real hyperbolic space) (Cannon et al., 1997) as frequency domain and spatial domain respectively. Then FFT and IFFT enable us to convert the embeddings between the two geometric spaces, accomplishing the leverage of real hyperbolic transformations to the complex hyperbolic model. The framework is simple and effective in learning the complex hyperbolic KG representations.

Figure 1 summarizes the comparison among embedding spaces for hierarchical patterns and multi-relation properties. In experiments, we evaluate our approach on the KG link prediction task with two popular benchmarks—WN18RR (Bordes et al., 2013) and FB15k-237 (Toutanova and Chen, 2015). Empirical results show that our Fourier transform-based complex hyperbolic KG embedding approach outperforms the baseline models in other geometric spaces.

The code and data of our work are available at `https://github.com/HKUST-KnowComp/ComplexHyperbolicKGE`.

2 Related Work

Euclidean KG embeddings. The traditional KG embedding models first started with the Euclidean geometry because of its convenient vectorial structure and closed-form computations such as distance formula and inner-product. After the occurrence of the translation-based models (Bordes et al., 2013) and bilinear models (Nickel et al., 2011; Yang et al., 2015), several extensions (Wang et al., 2014; Lin et al., 2015; Ji et al., 2015) have been made to further develop the Euclidean methods.

Complex Euclidean KG embeddings. The follow-up works (Trouillon et al., 2016; Hayashi and Shimbo, 2017; Sun et al., 2019) extended the traditional Euclidean models to complex hyperbolic geometry. Specifically, ComplEx (Trouillon et al., 2016) found that the Hermitian dot product can effectively capture anti-symmetric relations while retaining the efficiency benefits of the dot product. RotatE (Sun et al., 2019) defined each relation as a rotation in the complex vector space to infer various relation patterns (symmetry/anti-symmetry, inversion, composition). The effectiveness of these models revealed the potential of the complex geometry.

Hyperbolic embeddings. In recent years, the hyperbolic space attracted much attention for representation learning since it can naturally characterize tree structures. The hyperbolic embedding methods have developed from the single transitive relation graphs (Nickel and Kiela, 2017, 2018; Sonthalia and Gilbert, 2020) to multi-relational KGs (Balazevic et al., 2019; Chami et al., 2020). MurP (Balazevic et al., 2019) embedded the hierarchical multi-relational data in the Poincaré ball model and learned relation-specific parameters by Möbius operations. The state-of-the-art hyperbolic KG embedding models are a series of hyperbolic transformation-based models RefH, RotH, and AttH (Chami et al., 2020), which utilize the geometric tree-like property to capture the hierarchical structure naturally while using different geometric transformations as well as attention mechanism to parameterize other relation properties.

Lightweight Euclidean-based models. Based on the hyperbolic embedding model RotH (Chami et al., 2020), Wang et al. (2021) developed two lightweight Euclidean-based models RotL and Rot2L, which simplified the hyperbolic operations while keeping the flexible normalization effect.

Complex hyperbolic embeddings. Since many real-world hierarchically structured data such as taxonomies (Miller, 1995; Suchanek et al., 2007) and multitree networks (Griggs et al., 2012) have varying local structures, they do not ubiquitously match the hyperbolic geometry. Therefore, Xiao et al. (2021) explored the complex hyperbolic space to embed a variety of hierarchical structures. The complex hyperbolic embedding approach improved over the hyperbolic embedding models, but it only focused on the representation of single-relational graphs instead of multi-relational KGs.

Fourier Transform. Fourier transform (Heideman et al., 1984) converts a finite-sequence signal from its temporal or spatial domain to the frequency domain. FFT (Cooley et al., 1969) is a practical algorithm that computes the discrete Fourier transform (DFT) of a sequence. FFT and inverse FFT are widely used for many applications (Rockmore, 2000; Burgess, 2014) for their usefulness in signal processing as well as computation efficiency. They are also used to efficiently perform operations such as convolutions (Smith et al., 1997; Kipf and Welling, 2017) and cross-correlations (Bracewell and Bracewell, 1986; Wang et al., 2018). Hayashi and Shimbo (2019) also introduced the Fourier transform in KGE, where the main idea was to use the block circulant matrices to parameterize relations. While in our work, the Fourier transform is used to transform the entity embeddings between different geometric spaces.

3 Preliminaries

3.1 Hyperbolic Geometry

The hyperbolic space is a homogeneous space with constant negative curvature (Cannon et al., 1997). In the hyperbolic space, the volume of a ball grows exponentially with its radius. Contrastively, in the Euclidean space, the curvature is constantly 0, and the volume of a ball grows polynomially with its radius. The exponential volume growth rate enables the hyperbolic space to have powerful representation capability for tree structures since the number of nodes grows exponentially with the depth in a tree, while the Euclidean space is too flat and narrow to embed trees.

The Poincaré Ball Model. To describe the hyperbolic space in mathematical language, there are several models, among which the Poincaré ball model is popular for graph representation (Nickel and Kiela, 2017; Chami et al., 2020) due to the relatively convenient computations.

Denote the Poincaré ball model with constant negative curvature $-c$ as $\mathcal{P}_{\mathbb{R}}^{N} = \{\mathbf{x} \in \mathbb{R}^{N} : \|\mathbf{x}\|^2 < \frac{1}{c}\}$, which represents the open N-dimension ball in the ambient Euclidean space ($\|\cdot\|$ is the Euclidean L_2 norm). By the framework of gyrovector space (Ungar, 2008), the hyperbolic space can be formalized as an approximated vectorial structure, where the Möbius addition (Ganea et al., 2018) is used as the vector addition in $\mathcal{P}_{\mathbb{R}}^{N}$:

$$\mathbf{x} \oplus_c \mathbf{y} = \frac{(1 + 2c\mathbf{x}\mathbf{y} + c\|\mathbf{y}\|^2)\mathbf{x} + (1 - c\|\mathbf{x}\|^2)\mathbf{y}}{1 + 2c\mathbf{x}\mathbf{y} + c^2\|\mathbf{x}\|^2\|\mathbf{y}\|^2}. \quad (1)$$

Then the distance function in $\mathcal{P}_{\mathbb{R}}^{N}$ is given by

$$d_{\mathcal{P}}(\mathbf{x}, \mathbf{y}) = \frac{2}{\sqrt{c}} \operatorname{artanh}(\sqrt{c}\| - \mathbf{x} \oplus_c \mathbf{y}\|). \quad (2)$$

The practical computations in the hyperbolic space are often implemented using the tangent space. For $\mathbf{x} \in \mathcal{P}_{\mathbb{R}}^{N}$, the associated tangent space $\mathcal{T}_{\mathbf{x}}\mathcal{P}_{\mathbb{R}}^{N}$ is an N-dimension Euclidean space containing all tangent vectors passing through $\mathbf{x}$ (do Carmo, 1976). The manifold of the Poincaré ball model and the tangent space have closed-form maps to each other, which are defined as the exponential map $\exp_{\mathbf{0}}^{c}(\mathbf{v}) : \mathcal{T}_{\mathbf{0}}\mathcal{P}_{\mathbb{R}}^{N} \mapsto \mathcal{P}_{\mathbb{R}}^{N}$ and the logarithmic map $\log_{\mathbf{0}}^{c}(\mathbf{y}) : \mathcal{P}_{\mathbb{R}}^{N} \mapsto \mathcal{T}_{\mathbf{0}}\mathcal{P}_{\mathbb{R}}^{N}$:

$$\exp_{\mathbf{0}}^{c}(\mathbf{v}) = \tanh(\sqrt{c}\|\mathbf{v}\|)\frac{\mathbf{v}}{\sqrt{c}\|\mathbf{v}\|}, \quad (3)$$

$$\log_{\mathbf{0}}^{c}(\mathbf{y}) = \operatorname{artanh}(\sqrt{c}\|\mathbf{y}\|)\frac{\mathbf{y}}{\sqrt{c}\|\mathbf{y}\|}. \quad (4)$$

3.2 Complex Hyperbolic Geometry

The complex hyperbolic space is a homogeneous space of variable negative curvature (Goldman, 1999). The complex hyperbolic space also maintains the tree-like exponential volume growth property. From the properties of the complex hyperbolic geometry, we see that the complex hyperbolic space can naturally handle data with diverse local structures because of the non-constant curvature while preserving the tree-like properties to better capture the transitivity (Xiao et al., 2021).

The Unit Ball Model. The complex hyperbolic space's ambient Hermitian vector space $\mathbb{C}^{n,1}$ is the complex Euclidean space $\mathbb{C}^{n+1}$ endowed with some Hermitian form $\langle\!\langle \mathbf{z}, \mathbf{w} \rangle\!\rangle$, where $\mathbf{z}, \mathbf{w} \in \mathbb{C}^{n+1}$. Different choices of the Hermitian form $\langle\!\langle \mathbf{z}, \mathbf{w} \rangle\!\rangle$ correspond to different models of complex hyperbolic geometry, such as the unit ball model and the Siegel domain model (Parker, 2003). Here we choose the standard Hermitian form which derives the unit ball model:

$$\langle\!\langle \mathbf{z}, \mathbf{w} \rangle\!\rangle = z_1\overline{w_1} + \cdots + z_n\overline{w_n} - z_{n+1}\overline{w_{n+1}}, \quad (5)$$

where $\overline{w}$ denotes the complex conjugate of w. Then via the projective geometry (Goldman, 1999), the formula of the unit ball model is:[2]

$$\mathcal{B}^n_{\mathbb{C}} = \{(z_1, \cdots, z_n, 1) | |z_1|^2 + \cdots + |z_n|^2 < 1\}. \quad (6)$$

The metric on $\mathcal{B}^n_{\mathbb{C}}$ is Bergman metric, which takes the formula below in 2-d case:

$$ds^2 = \frac{-4}{\langle\!\langle \mathbf{z}, \mathbf{z} \rangle\!\rangle^2} \det \begin{bmatrix} \langle\!\langle \mathbf{z}, \mathbf{z} \rangle\!\rangle & \langle\!\langle d\mathbf{z}, \mathbf{z} \rangle\!\rangle \\ \langle\!\langle \mathbf{z}, d\mathbf{z} \rangle\!\rangle & \langle\!\langle d\mathbf{z}, d\mathbf{z} \rangle\!\rangle \end{bmatrix}. \quad (7)$$

Then the distance function on $\mathcal{B}^n_{\mathbb{C}}$ can be derived from the metric tensor:

$$d_{\mathcal{B}}(\mathbf{z}, \mathbf{w}) = \text{arcosh}\Big(2\frac{\langle\!\langle \mathbf{z}, \mathbf{w} \rangle\!\rangle \langle\!\langle \mathbf{w}, \mathbf{z} \rangle\!\rangle}{\langle\!\langle \mathbf{z}, \mathbf{z} \rangle\!\rangle \langle\!\langle \mathbf{w}, \mathbf{w} \rangle\!\rangle} - 1\Big). \quad (8)$$

Although the unit ball model and the Poincaré ball model look similar in mathematical formulae, they have many differences in properties since the complex hyperbolic geometry and hyperbolic geometry are intrinsically different geometries. Not only the variable/constant negative curvature but also their distance functions and the geometric computations vary with each other.

[2]Here we denote the dimension as n instead of N as in Section 3.1 since the dimensions of the two models can differ in Fourier transform, which we will see in Section 4.2.

4 Approach

Given the KG with entity set $V = \{e_j\}_{j=1}^m$, relation set $R = \{r_j\}_{j=1}^k$, and triplet set $F = \{(h, r, t) | h, t \in V, r \in R\}$, the link prediction task aims to predict the tail entity t for each test query (h, r). To train the models for a higher-quality inference, we learn the entity embeddings $\{\mathbf{e}_j\}_{j=1}^m$ in the unit ball model, while parameterizing the relations by the hyperbolic transformations RotH, RefH, and the hyperbolic attention-based model AttH (Chami et al., 2020). We construct the conversion between the two geometries through Fourier transform. Figure 2 presents the overview of our framework. In this section, we first present the attention-based hyperbolic transformations as relation parameterization, then introduce the Fourier transform as the conversion between complex hyperbolic domain and hyperbolic domain, followed by the details of our framework.

4.1 Relation Parameterization by Hyperbolic Transformations and Attention

In our work, we adopt the attention-based hyperbolic transformations developed by Chami et al. (2020) as the relation parameterization in the Poincaré ball model. Here we present the models RotH, RefH, and AttH.

RotH and RefH represent rotations and reflections in the hyperbolic space respectively. They can be modeled using the Givens transformation matrices, which take the following formula:

$$G^{\pm}(\theta) = \begin{bmatrix} \cos(\theta) & \mp\sin(\theta) \\ \sin(\theta) & \pm\cos(\theta) \end{bmatrix}. \quad (9)$$

Let $\Theta_r \equiv (\theta_{r,j})_{j\in\{1,\ldots,\frac{N}{2}\}}$ and $\Phi_r \equiv (\phi_{r,j})_{j\in\{1,\ldots,\frac{N}{2}\}}$ be the relation parameters, where N denotes the dimension of the embedding space, then the hyperbolic rotation and reflection are parameterized by the block-diagonal Givens matrices:

$$\text{Rot}(\Theta_r) = \text{diag}(G^+(\theta_{r,1}), \ldots, G^+(\theta_{r,\frac{N}{2}})), \quad (10)$$

$$\text{Ref}(\Phi_r) = \text{diag}(G^-(\phi_{r,1}), \ldots, G^-(\phi_{r,\frac{N}{2}})). \quad (11)$$

Thus, given a query (h, r), RotH and RefH apply hyperbolic rotation and reflection with relation-specific parameters $\mathbf{r}$ to the head embeddings $\tilde{\mathbf{h}} \in \mathcal{P}^N_{\mathbb{R}}$, and then get the query embeddings:

$$\text{RotH}(\tilde{\mathbf{h}}, \mathbf{r}) = \text{Rot}(\Theta_r)\tilde{\mathbf{h}}, \text{RefH}(\tilde{\mathbf{h}}, \mathbf{r}) = \text{Ref}(\Phi_r)\tilde{\mathbf{h}}. \quad (12)$$

In order to handle multiple relation properties,

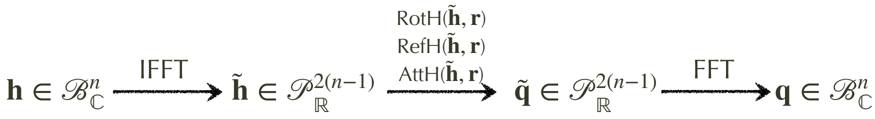

Figure 2: The inference process for a query (h, r) in our proposed complex hyperbolic KG embedding framework. $\mathbf{h}$ and $\tilde{\mathbf{h}}$ are the head entity embeddings in different spaces. $\mathbf{q}$ and $\tilde{\mathbf{q}}$ are the query embeddings in different spaces. $\mathcal{B}_{\mathbb{C}}^{n}$ denotes the n-dimension unit ball model in the complex hyperbolic space while $\mathcal{P}_{\mathbb{R}}^{2(n-1)}$ denotes the $2(n-1)$-dimension Poincaré ball model in the hyperbolic space.

AttH combines the above two representations using the hyperbolic attention and adding a hyperbolic translation $\mathbf{r}_r$ by Möbius addition (Eq. (1)):

$$\text{AttH}(\tilde{\mathbf{h}}, \mathbf{r}) = \text{Att}(\text{RotH}(\tilde{\mathbf{h}}, \mathbf{r}), \text{RefH}(\tilde{\mathbf{h}}, \mathbf{r}); \mathbf{a}_r) \oplus_{c_r} \mathbf{r}_r, \quad (13)$$

where c_r is the curvature parameter of r. RotH, RefH, and AttH leverage the trainable curvature so that each relation has its own curvature parameterization. The hyperbolic attention is constructed from the exponential map (Eq. (3)) of the average in the tangent space (Chami et al., 2019; Liu et al., 2019). More details about the hyperbolic attention mechanism can be referred to (Chami et al., 2020).

4.2 Conversion by Fourier Transform

The orthonormal Discrete Fourier Transform (DFT) $\mathcal{F}$ and its inverse (IDFT) $\mathcal{F}^{-1}$ between two finite complex-valued sequences $\{x_p\}_{p=0}^{N-1}$ and $\{z_q\}_{q=0}^{N-1}$ take the following formulae:

$$z_q = \mathcal{F}\{\mathbf{x}\}_q = \frac{1}{\sqrt{N}} \sum_{p=0}^{N-1} x_p \cdot e^{-i\frac{2\pi}{N}pq}, \quad (14)$$

$$x_p = \mathcal{F}^{-1}\{\mathbf{z}\}_p = \frac{1}{\sqrt{N}} \sum_{q=0}^{N-1} z_q \cdot e^{i\frac{2\pi}{N}pq}. \quad (15)$$

In our models, we transform the unit ball embeddings $\mathbf{z} \in \mathcal{B}_{\mathbb{C}}^{n}$ to the Poincaré ball embeddings $\mathbf{x} \in \mathcal{P}_{\mathbb{R}}^{N}$ back and forth. Note that the Poincaré ball embeddings $\mathbf{x} = \{x_0, \ldots, x_{N-1}\}$ are all real numbers, then $\mathcal{F}\{\mathbf{x}\}$ is symmetric: $z_q = \overline{z_{-q \bmod N}}, \forall q \in \{0, \ldots, N-1\}$. The dimension N is even because of the construction of diagonal Givens transformations (Eqs. (10) and (11)). Then it follows that z_0 and $z_{\frac{N}{2}}$ are real-valued, and the remainder of $\mathcal{F}\{\mathbf{x}\}$ is completely specified by just $\frac{N}{2} - 1$ complex numbers. Therefore, in practical algorithms, we set $N = 2(n-1)$, i.e., we use the first $\frac{N}{2} + 1$ elements $\{z_0, \ldots, z_{\frac{N}{2}}\}$ as the transformed unit ball embeddings.

We notice that the Fourier transform is not simply a conversion technique between complex and real domains. Performing circular convolutions in one domain equals the multiplication in another domain (Rader, 1972; Smith et al., 1997):

$$\begin{aligned}\{x \star y\}[n] &\triangleq \sum_{p=0}^{N-1} x_p \cdot y_{(n-p) \bmod N} \\ &= \mathcal{F}^{-1}\{\mathcal{F}\{\mathbf{x}\} \cdot \mathcal{F}\{\mathbf{y}\}\}_n. \quad (16)\end{aligned}$$

Its effectiveness provides useful transforms, while its practicability is guaranteed by FFT and IFFT, where the fast Fourier algorithms can reduce the computing complexity from $O(N^2)$ to $O(N \log N)$ (Cooley and Tukey, 1965).

4.3 Complex Hyperbolic Embeddings with Fourier Transform

For a query (h, r), Figure 2 briefly describes the inference process in our framework. The head embedding $\mathbf{h} \in \mathcal{B}_{\mathbb{C}}^{n}$ is in the n-dimension unit ball model. We apply inverse Fourier transform (Eq. (15)) to $\mathbf{h}$ and get the transformed head embeddings in the $2(n-1)$-dimension Poincaré ball model $\tilde{\mathbf{h}} \in \mathcal{P}_{\mathbb{R}}^{2(n-1)}$: $\tilde{\mathbf{h}} = \mathcal{F}^{-1}\{\mathbf{h}\}$.

Then we can apply RotH, RefH (Eq. (12)), and AttH (Eq. (13)) to get the query embedding $\tilde{\mathbf{q}}$:

$$\tilde{\mathbf{q}}_{\text{ModelH}} = \text{ModelH}(\tilde{\mathbf{h}}, \mathbf{r}) \quad (17)$$

where $\text{ModelH} = \{\text{RotH}, \text{RefH}, \text{AttH}\}$ represents the corresponding hyperbolic embedding model.

The query embedding $\tilde{\mathbf{q}}$ then gets transformed back to the unit ball model by Fourier transform (Eq. (14)): $\mathbf{q} = \mathcal{F}\{\tilde{\mathbf{q}}\}$.

Finally, we use the following score function (Balazevic et al., 2019) to measure the likelihood of a triplet (h, r, t):

$$s(h, r, t) = -d_{\mathcal{B}}(\mathbf{q}, \mathbf{t})^2 + b_h + b_t, \quad (18)$$

where $d_{\mathcal{B}}(\mathbf{q}, \mathbf{t})$ is the unit ball model distance (Eq. (8)) between the tail embeddings $\mathbf{t}$ and the query embeddings $\mathbf{q}$ computed by the above procedures. b_h and b_t are bias terms of the head and tail entity. The model learns the embeddings by maximizing the score functions of the training triplets, i.e.,

	$\lvert V \rvert$	$\lvert R \rvert$	$\lvert F \rvert$	ξ_G
WN18RR	40,943	11	93,003	-2.54
FB15k-237	14,541	237	310,079	-0.65

Table 1: Data statistics. $|V|$, $|R|$, $|F|$ denote # entities, # relations, # triplets. ξ_G is the global graph curvature.

making the query embeddings of (h, r) close with its ground truth tail embeddings. The score function is also used to predict the test data.

In summary, our model parameters include the entity parameters: $\{\mathbf{e}_j\}_{j=1}^m \in \mathcal{B}_{\mathbb{C}}^n$ (embeddings), $\{b_j\}_{j=1}^m$ (biases); and the relation parameters: Θ_r (rotations), Φ_r (reflections), $\mathbf{r}_r$ (translations), $\mathbf{a}_r$ (attention), c_r (curvature). The FFT and IFFT can be computed very efficiently, so our models have almost the same computation cost with the base models RotH, RefH, and AttH, while we utilize a more powerful representation geometry to improve the embedding quality.

5 Experiments

In this section, we evaluate our approaches on the KG link prediction task. We show that our complex hyperbolic embedding models outperform the baseline methods based on other geometric spaces.

5.1 Experimental Settings

5.1.1 Data

We use two widely-used KG benchmarks to evaluate the embedding models. The data statistics are provided in Table 1. The global graph curvature ξ_G (Gu et al., 2019) is provided in (Chami et al., 2020), which is a distance-based measure to estimate the tree-likeness of graphs. A lower ξ_G corresponds to a more tree-like graph.

WN18RR. WN18RR (Bordes et al., 2013) is a knowledge graph dataset created from WN18, which is a subset of WordNet (Miller, 1995). WordNet is a large lexical database with hypernymy relation, so WN18RR inherits the underlying hierarchical structure.

FB15k-237. FB15k-237 (Toutanova and Chen, 2015) is a knowledge graph dataset created from FB15k, which is derived from Freebase. Compared with WN18RR, FB15k-237 has much more relations and various relation properties, resulting in a more flexible structure, which can be reflected by the larger ξ_G.

We follow the train-valid-test data splitting of previous works (Chami et al., 2020; Wang et al., 2021), where # train-valid-test triplets are $86,845$-$3,034$-$3,134$ for WN18RR and $272,115$-$17,535$-$20,466$ for FB15k-237. The data can be obtained in the public repository of (Chami et al., 2020).[3]

5.1.2 Baselines

The following KG embedding baselines are compared with our approaches (**FFTRefH**, **FFTRotH**, **FFTAttH**): complex Euclidean embedding models **ComplEx-N3** (Lacroix et al., 2018) and **RotatE** (Sun et al., 2019); hyperbolic embedding models **MuRP** (Balazevic et al., 2019), **RefH**, **RotH**, **AttH** (Chami et al., 2020); the Euclidean analogues of the hyperbolic methods **MuRE**, **RefE**, **RotE**, **AttE**; the lightweight Euclidean-based models **RotL**, **Rot2L** (Wang et al., 2021).

5.1.3 Training and Evaluation

For the baselines, we either take the results from the original papers (Chami et al., 2020; Wang et al., 2021) (Table 2) or use their released best hyperparameters as well as their open-source codes to train their models (Table 3, 4, and 5). For our approaches, we tune the hyperparameters by grid search on each validation set in 32-dimension complex hyperbolic space, which are given in Appendix A. Our embedding models are trained by optimizing the full cross-entropy loss with uniform negative sampling. We conduct all the experiments on four NVIDIA GTX 1080Ti GPUs with 11GB memory each.

We use the mean reciprocal rank (MRR) and the proportion of correct types that rank no larger than N (Hits@N) as our evaluation metrics, which are widely used for evaluating link prediction. We follow the filtered evaluation setting (Bordes et al., 2013) to filter out the true triplets during evaluation. In all experiments, each running is executed five times and the mean values of results are reported.

5.2 Overall Results

Table 2 presents the results in 32-dimension embedding spaces. We strictly follow the experimental setting and data splitting of the previous works (Chami et al., 2020; Wang et al., 2021). The results of the baselines are taken from the original papers, where RotL and Rot2L do not report the Hits@3 scores, thus we leave them blank.

[3] https://github.com/HazyResearch/KGEmb.

		WN18RR				FB15k-237			
Geometry	Model	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10
$\mathbb{C}^n$	ComplEx-N3	0.420	0.390	0.420	0.460	0.294	0.211	0.322	0.463
	RotatE	0.387	0.330	0.417	0.491	0.290	0.208	0.316	0.458
$\mathbb{R}^n$	MuRE	0.458	0.421	0.471	0.525	0.313	0.226	0.340	0.489
	RefE	0.455	0.419	0.470	0.521	0.302	0.216	0.330	0.474
	RotE	0.463	0.426	0.477	0.529	0.307	0.220	0.337	0.482
	AttE	0.456	0.419	0.471	0.526	0.311	0.223	0.339	0.488
	RotL	0.469	0.426	-	0.550	0.320	0.229	-	0.500
	Rot2L	0.475	0.434	-	0.554	0.326	0.237	-	0.503
$\mathcal{P}^n_{\mathbb{R}}$	MuRP	0.465	0.420	0.484	0.544	0.323	0.235	0.353	0.501
	RefH	0.447	0.408	0.464	0.518	0.312	0.224	0.342	0.489
	RotH	0.472	0.428	0.490	0.553	0.314	0.223	0.346	0.497
	AttH	0.466	0.419	0.484	0.551	0.324	0.236	0.354	0.501
$\mathcal{B}^n_{\mathbb{C}}$	FFTRefH	0.463	0.412	0.480	0.547	0.325	0.234	0.359	0.508
	FFTRotH	**0.484**	**0.437**	**0.502**	**0.572**	0.319	0.228	0.352	0.500
	FFTAttH	0.476	0.432	0.494	0.558	**0.331**	**0.239**	**0.365**	**0.517**

Table 2: Evaluation of link prediction task in 32-dimension embedding spaces. The best results are shown in boldface. The second best results are underlined.

Relation	Khs_G	ξ_G	# Triplets	RefH	RotH	AttH	FFTRefH	FFTRotH	FFTAttH
member meronym	1.00	-2.90	253	0.316	0.383	0.383	0.366	**0.411**	0.402
hypernym	1.00	-2.46	1,251	0.218	0.268	0.257	0.249	**0.283**	0.268
has part	1.00	-1.43	172	0.259	0.303	0.294	0.287	**0.347**	0.335
instance hypernym	1.00	-0.82	122	0.471	0.480	0.471	0.496	**0.503**	0.499
member of domain region	1.00	-0.78	26	0.417	0.417	0.404	**0.436**	0.423	0.410
member of domain usage	1.00	-0.74	24	0.424	0.451	0.445	0.431	**0.458**	0.424
synset domain topic of	0.99	-0.69	114	0.352	0.417	0.406	0.436	**0.475**	0.444
derivationally related form	0.07	-3.84	1,074	0.960	0.964	0.965	0.968	**0.969**	0.967
also see	0.36	-2.09	56	0.664	0.640	0.649	**0.684**	0.675	0.676
similar to	0.07	-1.00	3	**1.000**	**1.000**	0.944	**1.000**	**1.000**	**1.000**
verb group	0.07	-0.50	39	**0.974**	**0.974**	0.970	**0.974**	**0.974**	0.970

Table 3: Results of Hits@10 for WN18RR relations in 32-dimension embedding spaces. Higher Khs_G and lower ξ_G correspond to more tree-like. # Triplets means the triplet count of each relation in test set. The best results are shown in boldface.

The results show that our Fourier transform-based complex hyperbolic approaches have the best performance on the link prediction task, demonstrating the powerful representation capacity of the complex hyperbolic geometry and the effectiveness of Fourier transform. Specifically, FFTRotH achieves the best results on WN18RR, while FFTAttH outperforms other methods on FB15k-237. The relations in WN18RR typically have transitivity property, in which case the hyperbolic rotation takes more advantages. FB15k-237 is a more challenging link prediction dataset since it has more relations and varying structures as well as a larger scale of triplets. Therefore, the attention mechanism helps to generalize the hyperbolic transformations to multiple relation properties.

From Table 2, we see that the traditional complex Euclidean models (ComplEx-N3 and RotatE) do not have competitive performance with the hyperbolic KG embedding models or their Euclidean analogues. The hyperbolic methods (MuRP, RefH, RotH, and AttH) have better results than their Euclidean analogues (MuRE, RefE, RotE, and AttE), revealing the improvements of the hyperbolic geometry over Euclidean geometry in low-dimensional KG representation. RotL replaced the Möbius addition of RotH with a new flexible addition operation, while Rot2L further utilizes two stacked rotation-translation layers in the Euclidean space. The two Euclidean-based methods outperform their base model RotH by adapting a lightweight architecture. However, they still cannot achieve as promising results as the complex hyperbolic embedding approaches.

	8-dimension		16-dimension		32-dimension		64-dimension	
Model	MRR	Hits@1	MRR	Hits@1	MRR	Hits@1	MRR	Hits@1
RefH	0.190	0.140	0.401	0.360	0.447	0.408	0.475	0.433
RotH	0.220	0.154	0.417	0.370	0.472	0.428	**0.488**	**0.442**
AttH	0.158	0.102	0.404	0.356	0.466	0.419	0.476	0.430
FFTRefH	0.369	0.319	0.447	0.408	0.463	0.412	0.469	0.425
FFTRotH	**0.411**	**0.358**	**0.468**	**0.423**	**0.484**	**0.437**	**0.488**	**0.442**
FFTAttH	0.387	0.330	0.459	0.415	0.476	0.432	0.479	0.435

Table 4: Results of MRR and Hits@1 in different embedding dimensions on WN18RR. The best results are shown in boldface.

	8-dimension		16-dimension		32-dimension		64-dimension	
Model	MRR	Hits@1	MRR	Hits@1	MRR	Hits@1	MRR	Hits@1
RefH	0.267	0.188	0.288	0.204	0.312	0.224	0.328	0.237
RotH	0.269	0.187	0.289	0.204	0.314	0.223	0.323	0.231
AttH	0.276	0.194	0.298	0.212	0.324	0.236	0.333	0.240
FFTRefH	0.281	0.198	0.304	0.217	0.325	0.234	0.337	0.242
FFTRotH	0.287	0.201	0.306	0.217	0.319	0.228	0.323	0.231
FFTAttH	**0.295**	**0.209**	**0.314**	**0.224**	**0.331**	**0.239**	**0.339**	**0.245**

Table 5: Results of MRR and Hits@1 in different embedding dimensions on FB15k-237. The best results are shown in boldface.

5.3 Exploring the Relations

In Section 5.2, we see that the overall results of Fourier transform-based complex hyperbolic methods surpass their corresponding hyperbolic methods. Here we explore their performance on each relation of WN18RR. For each relation, we give their statistics of Krackhardt hierarchy score (Khs_G) (Balazevic et al., 2019) and estimated graph curvature ξ_G (Chami et al., 2019). Higher Khs_G and lower ξ_G mean more tree-like, i.e., the relation is more transitive. We report the Hits@10 scores in Table 3.

We find that for most relations, FFT complex hyperbolic methods outperform hyperbolic methods significantly. For the transitive relations such as *member meronym*, *hypernym*, *has part*, etc, rotation has much better results than reflection. This phenomenon is consistent with the analysis of previous work (Chami et al., 2020), where they found hyperbolic rotations work better on anti-symmetric relations while hyperbolic reflections encode symmetric relations better. Transitivity fulfills anti-symmetry naturally, so rotation gains higher scores (RotH>RefH, FFTRotH>FFTRefH). For the symmetric relation such as *also see*, reflection outperforms rotation (RefH>RotH, FFTRefH>FFTRotH). Since most relations in WN18RR exhibit transitivity, the rotation models have better performance than the reflection models in overall results (Table 2). Regardless of the relation properties, our approaches improve the corresponding hyperbolic methods largely, except for the relations with few test triplets such as *similar to* and *verb group*, where they all have close-to-1 Hits@10 results.

5.4 Exploring the Embedding Dimensions

In this section, we explore the performance of Fourier transform-based complex hyperbolic approaches and the corresponding hyperbolic methods in various embedding dimensions. The results are presented in Table 4 and 5. We find that when the embedding dimension is small, the complex hyperbolic approaches outperform the hyperbolic base models by a large margin. Remarkably, FFTRotH improves over RotH by around 100% in 8-dimension on WN18RR. With the increase of the embedding dimension, their predictions get more and more similar and gradually converge. The results reveal the effectiveness of our approaches especially in small dimensions, demonstrating the strong representation capacity of complex hyperbolic geometry.

6 Conclusion and Future Work

In this work, we explore the complex hyperbolic geometry for multi-relational KG embeddings. The whole framework utilizes the Fourier transform as the efficient conversion between geometric spaces. With the aid of the Fourier transform, the complex hyperbolic embeddings can be transformed into the real domain and be capable of applying real hyperbolic transformations, which enables our approach to take the advantages of both the powerful complex hyperbolic geometry and the attention-based real hyperbolic transformations. Experiments show that the Fourier transform-based complex hyperbolic embedding models can effectively learn the KG embeddings and outperform the baseline models of other spaces in the link prediction task. We believe our proposed approach not only provides a novel and interesting representation learning framework for KGs but also potentially inspires the learning algorithms for more general multi-relational data and contributes to improvements on more downstream tasks.

Limitations

Limited improvements in high dimensions. Although our approaches can significantly outperform the baselines in low-dimensional KG embedding setting, we find that our approaches would get converge and have close results with the hyperbolic base models in sufficiently high dimensions. For example, in Table 4, FFTRotH and RotH have the same results in 64-dimension embedding spaces on WN18RR.

This issue has been observed previously (Nickel and Kiela, 2017; Chami et al., 2020), though their comparisons are established between hyperbolic space and Euclidean space. The representation capacity gap between geometric spaces is distinctly revealed in low dimensions. The gap may get eliminated to some extent by increasing the dimension. The complex hyperbolic geometry and hyperbolic geometry usually converge their results in much lower dimensions than Euclidean geometry because of the exponential growth property, resulting in the limited improvements in high dimensions.

Acknowledgements

The authors of this paper were supported by the NSFC Fund (U20B2053) from the NSFC of China, the RIF (R6020-19 and R6021-20) and the GRF (16211520) from RGC of Hong Kong, the MHKJFS (MHP/001/19) from ITC of Hong Kong and the National Key R&D Program of China (2019YFE0198200) with special thanks to HKMAAC and CUSBLT, and the Jiangsu Province Science and Technology Collaboration Fund (BZ2021065). We also thank the support from the UGC Research Matching Grants (RMGS20EG01-D, RMGS20CR11, RMGS20CR12, RMGS20EG19, RMGS20EG21). We also thank the support from NVIDIA AI Technology Center (NVAITC).

References

Ivana Balazevic, Carl Allen, and Timothy M. Hospedales. 2019. Multi-relational poincaré graph embeddings. In *NeurIPS*, pages 4465–4475.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *NIPS*, pages 2787–2795.

Ronald Newbold Bracewell and Ronald N Bracewell. 1986. *The Fourier transform and its applications*, volume 31999. McGraw-Hill New York.

Richard James Burgess. 2014. *The history of music production*. Oxford University Press.

James W Cannon, William J Floyd, Richard Kenyon, Walter R Parry, et al. 1997. Hyperbolic geometry. *Flavors of geometry*, 31(59-115):2.

Ines Chami, Adva Wolf, Da-Cheng Juan, Frederic Sala, Sujith Ravi, and Christopher Ré. 2020. Low-dimensional hyperbolic knowledge graph embeddings. In *ACL*, pages 6901–6914. Association for Computational Linguistics.

Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. 2019. Hyperbolic graph convolutional neural networks. In *NeurIPS*, pages 4869–4880.

James W Cooley and John W Tukey. 1965. An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, 19(90):297–301.

JW Cooley, P Lewis, and P Welch. 1969. The finite fourier transform. *IEEE Transactions on audio and electroacoustics*, 17(2):77–85.

Manfredo P. do Carmo. 1976. *Differential geometry of curves and surfaces*. Prentice Hall.

Octavian-Eugen Ganea, Gary Bécigneul, and Thomas Hofmann. 2018. Hyperbolic neural networks. In *NeurIPS*, pages 5350–5360.

William Mark Goldman. 1999. *Complex hyperbolic geometry*. Oxford University Press.

Jerrold R. Griggs, Wei-Tian Li, and Linyuan Lu. 2012. Diamond-free families. *J. Comb. Theory, Ser. A*, 119(2):310–322.

Albert Gu, Frederic Sala, Beliz Gunel, and Christopher Ré. 2019. Learning mixed-curvature representations in product spaces. In *ICLR (Poster)*. OpenReview.net.

Katsuhiko Hayashi and Masashi Shimbo. 2017. On the equivalence of holographic and complex embeddings for link prediction. In *ACL (2)*, pages 554–559. Association for Computational Linguistics.

Katsuhiko Hayashi and Masashi Shimbo. 2019. A non-commutative bilinear model for answering path queries in knowledge graphs. In *EMNLP/IJCNLP (1)*, pages 2422–2430. Association for Computational Linguistics.

Michael Heideman, Don Johnson, and Charles Burrus. 1984. Gauss and the history of the fast fourier transform. *IEEE ASSP Magazine*, 1(4):14–21.

Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Knowledge graph embedding via dynamic mapping matrix. In *ACL (1)*, pages 687–696. The Association for Computer Linguistics.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *ICLR (Poster)*. OpenReview.net.

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

Timothée Lacroix, Nicolas Usunier, and Guillaume Obozinski. 2018. Canonical tensor decomposition for knowledge base completion. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pages 2869–2878. PMLR.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *AAAI*, pages 2181–2187. AAAI Press.

Qi Liu, Maximilian Nickel, and Douwe Kiela. 2019. Hyperbolic graph neural networks. In *NeurIPS*, pages 8228–8239.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

Maximilian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In *NIPS*, pages 6338–6347.

Maximilian Nickel and Douwe Kiela. 2018. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pages 3776–3785. PMLR.

Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. 2016. A review of relational machine learning for knowledge graphs. *Proc. IEEE*, 104(1):11–33.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A three-way model for collective learning on multi-relational data. In *ICML*, pages 809–816. Omnipress.

John R Parker. 2003. Notes on complex hyperbolic geometry. *preprint*.

Charles M. Rader. 1972. Discrete convolutions via mersenne transrorms. *IEEE Transactions on Computers*, 21(12):1269–1273.

Daniel N. Rockmore. 2000. The FFT: an algorithm the whole family can use. *Comput. Sci. Eng.*, 2(1):60–64.

Steven W Smith et al. 1997. The scientist and engineer's guide to digital signal processing.

Rishi Sonthalia and Anna C. Gilbert. 2020. Tree! I am no tree! I am a low dimensional hyperbolic embedding. In *NeurIPS*.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *WWW*, pages 697–706. ACM.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. In *ICLR (Poster)*. OpenReview.net.

Kristina Toutanova and Danqi Chen. 2015. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd workshop on continuous vector space models and their compositionality*, pages 57–66.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *ICML*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 2071–2080. JMLR.org.

Abraham Albert Ungar. 2008. *Analytic hyperbolic geometry and Albert Einstein's special theory of relativity*. World Scientific.

Chen Wang, Le Zhang, Lihua Xie, and Junsong Yuan. 2018. Kernel cross-correlator. In *AAAI*, pages 4179–4186. AAAI Press.

Kai Wang, Yu Liu, Dan Lin, and Michael Sheng. 2021. Hyperbolic geometry is not necessary: Lightweight euclidean-based models for low-dimensional knowledge graph embeddings. In *EMNLP (Findings)*, pages 464–474. Association for Computational Linguistics.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *AAAI*, pages 1112–1119. AAAI Press.

Huiru Xiao, Caigao Jiang, Yangqiu Song, James Zhang, and Junwu Xiong. 2021. Unit ball model for embedding hierarchical structures in the complex hyperbolic space. *arXiv preprint arXiv:2105.03966*.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *ICLR (Poster)*.

A Hyperparameters

We tune our hyperparameters by grid search on each validation set in 32-dimension complex hyperbolic space, which are given in Table 6. For FFT and IFFT algorithms, we use the package torch.fft[4] and set the parameter norm="ortho", which is consistent with the defined orthonormal Fourier transform in Section 4.2.

[4]https://pytorch.org/docs/stable/fft.html.

Data	Model	Optimizer	Batch size	Negative samples	Learning rate	Double negative
WN18RR	FFTRefH	Adam	500	100	0.0003	True
	FFTRotH	Adam	500	100	0.0003	True
	FFTAttH	Adam	500	100	0.0004	True
FB15k-237	FFTRefH	Adagrad	500	250	0.02	False
	FFTRotH	Adam	100	100	0.0002	False
	FFTAttH	Adagrad	500	100	0.03	False

Table 6: Hyperparameters of our approaches.