3D Rotation and Translation for Hyperbolic Knowledge Graph Embedding

Anonymous ACL submission

Abstract

The main objective of Knowledge Graph (KG) embeddings is to learn low-dimensional representations of entities and relations, enabling the prediction of missing facts. A significant challenge in achieving better KG embeddings lies in capturing relation patterns, including symmetry, antisymmetry, inversion, commutative composition, non-commutative composition, hierarchy, and multiplicity. This study introduces a novel model called 3H-TH (3D Rotation and Translation in Hyperbolic space) that captures these relation patterns simultaneously. In contrast, previous attempts have not achieved satisfactory performance across all the mentioned properties at the same time. The experimental results demonstrate that the new model outperforms existing state-of-the-art models in terms of accuracy, hierarchy property, and other relation patterns in low-dimensional space, meanwhile performing similarly in high-dimensional space.

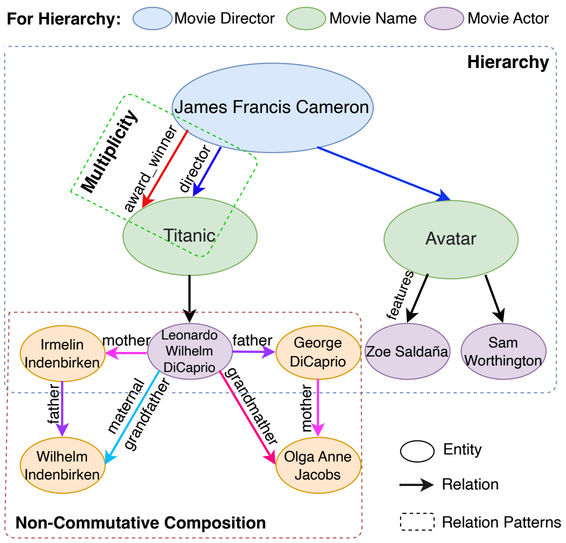

Figure 1: Toy examples for three difficult relation patterns. Our approach can perform well in Hierarchy, Multiplicity, and Non-Commutative Composition.

1 Introduction

The components of a knowledge graph are collections of factual triples, where each triple (h, r, t) denotes a relation r between a head entity h and a tail entity t; toy examples are shown in Fig. 1. Freebase (Bollacker et al., 2008), Yago (Suchanek et al., 2007), and WordNet (Miller, 1995) are some examples of knowledge graphs used in the real world. Meanwhile, applications such as question-answering (Hao et al., 2017), information retrieval (Xiong et al., 2017), recommender systems (Zhang et al., 2016), and natural language processing (Yang and Mitchell, 2019) may find significant value for knowledge graphs. Therefore, knowledge graph research is receiving increasing attention in both the academic and business domains.

Predicting missing links is a crucial aspect of knowledge graphs, given their typical incompleteness. In recent years, significant research efforts have focused on addressing this challenge through the utilization of knowledge graph embedding (KGE) techniques, which involve learning low-dimensional representations of entities and relations (Bordes et al., 2013; Trouillon et al., 2016). KGE approaches have demonstrated scalability and efficiency in modeling and inferring knowledge graph entities and relations based on available facts.

A major issue in KGE research concerned several relation patterns, including symmetry, antisymmetry, inversion, composition (i.e., commutative and non-commutative composition), hierarchy, and multiplicity (see Appendix A.8). In fact, several current approaches have attempted to model one or more of the above relation patterns (Bordes et al., 2013; Sun et al., 2019; Chami et al., 2020; Cao et al., 2021). The TransE (Bordes et al., 2013), which models the antisymmetry, inversion, and composition patterns, represents relations as translations. The RotatE (Sun et al., 2019) represents

Method	Symmetry	Antisymmetry	Inversion	Commutative	Non-commutative	Hierarchy	Multiplicity
TransE (TE)		✓	✓	✓			
RotatE (2E)	✓	✓	✓	✓			
QuatE (3E)	✓	✓	✓	✓	✓		
MuRP (TH)		✓	✓	✓		✓	
RotH (2H)	✓	✓	✓	✓		✓	✓
DualE	✓	✓	✓	✓	✓		✓
(Proposal) 3H-TH	✓	✓	✓	✓	✓	✓	✓

Table 1: Relation patterns for existing and proposed models (✓means "can")

the relation as a rotation and aims to model symmetry, antisymmetry, inversion, and composition. For some difficult patterns (see Fig. 1), including non-commutative composition, hierarchy, and multiplicity, the AttH (Chami et al., 2020) embeds relation in hyperbolic space to enable relations to acquire hierarchy property. The DualE (Cao et al., 2021) attempts to combine translation and rotation operations to model multiple relations. Such approaches, however, have failed to perform well on all the above relation patterns simultaneously as shown in Table 1. Our proposed method 3H-TH, meaning 3D rotation in hyperbolic space and translation in hyperbolic space, can simultaneously model these relation patterns.

Here we present how our proposed method (3H-TH) works for the difficult relation pattern examples in Fig. 1. By embedding the entities and relations in hyperbolic space, we can allow the KG model to acquire hierarchy properties so that we can more clearly distinguish between the different hierarchies of entities, for example, movie director, name, and actor. Besides, to solve non-commutative problems, for example (see Fig. 1), if the mother of A's father (B) is C while the father of A's mother (D) is E, then C and E are equal if the relations were commutative, we use the quaternion geometry property (non-commutative) to enable the model to obtain a non-commutative composition pattern. Finally, we try to combine rotation and translation operations to obtain multiplicity properties, e.g. different relations exist between the same entities (e.g., *award-winner, director*).

Moreover, our study provides some important insights into developing several comparable methods to explore the impact of a combination of translation and rotation in Euclidean or hyperbolic space, as well as both simultaneously. We evaluate the new model on three KGE datasets including WN18RR (Dettmers et al., 2018), FB15K-237 (Toutanova and Chen, 2015), and FB15K (Bordes et al., 2013). Experimental results show that the new model outperforms existing state-of-the-art models in terms of accuracy, hierarchy property, and other relation patterns in low-dimensional space, meanwhile performing similarly in high-dimensional space, which indicates that the new model 3H-TH can simultaneously model symmetry, antisymmetry, inversion, composition, hierarchy, and multiplicity relation patterns.

2 Related Work

Knowledge graph embedding has received a lot of attention from researchers in recent years. One of the main KGE directions has been led by translation-based and rotation-based approaches. Another key area is hyperbolic KGE, which enables models to acquire hierarchy property. In particular, our approach advances in both directions and acquires both advantages.

Translation-based approach. One of the widely adopted methods in KGE is the translation-based approach, exemplified by TransE (Bordes et al., 2013), which represents relation vectors as translations in the vector space. In this approach, the relationship between the head and tail entities is approximated by adding the relation vector to the head entity vector, resulting in a representation that is expected to be close to the tail entity vector. After TransE, there has been an increasing amount of literature on its extension. TransH (Wang et al., 2014) represents a relation as a hyperplane to help the model perform better on complex relations. By embedding entities and relations in separate spaces with a shared projection matrix, TransR (Lin et al., 2015) further creates a relation-specific space to obtain a more expressive model for different types of entities and relations. Compared to TransR, TransD (Ji et al., 2015) employs independent projection vectors for each object and relation, which can reduce the amount of computation. Although these methods are relatively simple and have only a few parameters, they do not effectively express crucial relation patterns such as symmetry, hierarchy, and multiplicity relations (Table 1).

Rotation-based approach. RotatE (Sun et al., 2019) introduced a new direction as rotation-based methods, which represents the relation vectors as rotation in complex vector space and can model various relation patterns, including symmetry, antisymmetry, inversion, and composition. QuatE (Zhang et al., 2019) substitutes 2D rotation with quaternion operation (3D rotation) in quaternion space, aiming to obtain a more expressive model than RotatE. Furthermore, the incorporation of 3D rotation enables the model to capture the non-commutative composition of relations, leveraging the geometric properties of quaternions (wherein two 3D rotations are known to be non-commutative). However, these rotation operations cannot solve hierarchy and multiplicity (Table 1). DualE (Cao et al., 2021) presents a solution to the multiplicity problem by combining translation and rotation operations. However, the experimental results discussed in this paper do not provide conclusive evidence of the model's effectiveness in handling multiple relation data.

Hyperbolic KGE. One of the major challenges for KGE is the hierarchy problem. Hyperbolic geometry has been shown to provide an efficient approach to representing KG entities and relations in low-dimensional space while maintaining latent hierarchy properties. MuRP (Balazevic et al., 2019) optimizes the hyperbolic distance between the projected head entity and the translational tail entity to achieve comparable results by using fewer dimensions than the previous methods. RotH (Chami et al., 2020) tries to substitute translation operations with rotation operations to obtain more relation patterns properties like RotatE. However, there is still room for improvement in handling other relation patterns, particularly in terms of multiplicity and non-commutative composition properties. BiQUE(Guo and Kok, 2021) utilizes biquaternions, which encompass both circular rotations in Euclidean space and hyperbolic rotations, aim to acquire hierarchy properties and RotatE-based relation patterns, while this approach struggles to effectively capture the Multiplicity property. Our proposed model 3H-TH leverages translation, 3D rotation, and hyperbolic embedding to offer a comprehensive and expressive representation of entities and relations, encompassing various relation patterns (Table 1).

3 Problem Formulation and Background

We describe the KGE problem and present some related methods before our approach part.

3.1 Knowledge graph embedding

Given a knowledge graph with a set of fact triples $(h, r, t) \in \mathcal{E} \subseteq \mathcal{V} \times \mathcal{R} \times \mathcal{V}$, where $\mathcal{V}$ and $\mathcal{R}$ represent sets of entities and relations, respectively. Mapping entities $v \in \mathcal{V}$ to embeddings $\mathbf{e}_v$ in $k_{\mathcal{V}}$ dimensions and relations $r \in \mathcal{R}$ to embeddings $\mathbf{e}_r$ in $k_{\mathcal{R}}$ dimensions is the goal of KGE.

We use the scoring function $s : \mathcal{V} \times \mathcal{R} \times \mathcal{V} \rightarrow \mathbb{R}$ to measure the difference between the transformed entities and target entities, and the difference is mainly composed of distance including Euclidean distance:

$$d^E(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|$$

and hyperbolic distance (Ganea et al., 2018):

$$d^{\xi_r}(\mathbf{x}, \mathbf{y}) = \frac{2}{\sqrt{\xi_r}} \tanh^{-1}(\sqrt{\xi_r}|| - \mathbf{x} \oplus^{\xi_r} \mathbf{y}||), \tag{1}$$

where $\|\cdot\|$, $\oplus^{\xi_r}$, and ξ_r represent L2 norm, Möbius addition (see Equation 11), and curvature in hyperbolic space, respectively.

3.2 TransE

Inspired by word2vec (Mikolov et al., 2013) in the domain of word embedding, TransE (Bordes et al., 2013) is the first translation-based work in the field of KGE, representing relations as translations in Euclidean space. Given triple vectors ($\mathbf{e}_h \in \mathbb{R}^k, \mathbf{e}_r \in \mathbb{R}^k, \mathbf{e}_t \in \mathbb{R}^k$), the scoring function of TransE is

$$s = -d^E(\mathbf{e}_h + \mathbf{e}_r, \mathbf{e}_t),$$

then maximize s to train this model.

3.3 2D and 3D rotation

To enable KGE models to acquire more relation patterns, including symmetry, antisymmetry, inversion, and composition, RotatE (Sun et al., 2019) represents relation as 2D rotation in complex space $\mathbb{C}$. Given triple vectors ($\mathbf{e}_h \in \mathbb{R}^k, \mathbf{c}_r \in \mathbb{C}^{\frac{k}{2}}, \mathbf{e}_t \in \mathbb{R}^k$), the scoring function of RotatE is

$$s = -d^E(\mathbf{e}_h \circ \mathbf{c}_r, \mathbf{e}_t),$$

where the elements of $\mathbf{c}_r$ are constrained to be on the unit circle in $\mathbb{C}$, i.e., $|(\mathbf{c}_r)_i| = 1$, and the symbol $\circ$ denotes Hadamard product.

Model	Relation embeddings	Translation	Rotation	Scoring function
TransE (TE)	$\mathbf{e}_r$	E		$-d^E(\mathbf{e}_h+\mathbf{e}_r,\mathbf{e}_t)+b_h+b_t$
RotatE (2E)	$\mathbf{c}_r$		2D in E	$-d^E(\mathbf{e}_h\circ\mathbf{c}_r,\mathbf{e}_t)+b_h+b_t$
QuatE (3E)	$\mathbf{q}_r$		3D in E	$(\mathbf{e}_h\otimes\mathbf{q}_r^{\triangleright})\cdot\mathbf{e}_t+b_h+b_t$
MuRP (TH)	$\mathbf{b}_r$	H		$-d^{\xi_r}\left(\mathbf{b}_h\oplus^{\xi_r}\mathbf{b}_r,\mathbf{b}_t\right)^2+b_h+b_t$
RotH (2H)	$\mathbf{c}_r$		2D in H	$-d^{\xi_r}(\mathbf{b}_h\circ\mathbf{c}_r,\mathbf{b}_t)^2+b_h+b_t$
3H	$\mathbf{q}_r$		3D in H	$-d^{\xi_r}(\mathbf{b}_h\otimes\mathbf{q}_r,\mathbf{b}_t)^2+b_h+b_t$
2E-TE	$\mathbf{c}_r,\mathbf{e}_r$	E	2D in E	$-d^E(\mathbf{e}_h\circ\mathbf{c}_r+\mathbf{e}_r,\mathbf{e}_t)+b_h+b_t$
3E-TE	$\mathbf{q}_r,\mathbf{e}_r$	E	3D in E	$-d^E(\mathbf{e}_h\otimes\mathbf{q}_r^{\triangleright}+\mathbf{e}_r,\mathbf{e}_t)+b_h+b_t$
2E-TE-2H-TH	$\mathbf{c}_{(r,E)},\mathbf{e}_r,\mathbf{c}_{(r,H)},\mathbf{b}_r$	E, H	2D in E, H	$-d^{\xi_r}\left((\mathbf{b}_\gamma\circ\mathbf{c}_{(r,H)})\oplus^{\xi_r}\mathbf{b}_r,\mathbf{b}_t\right)^2+b_h+b_t$
3H-TH	$\mathbf{q}_r,\mathbf{b}_r$	H	3D in H	$-d^{\xi_r}\left((\mathbf{b}_h\otimes\mathbf{q}_r^{\triangleright})\oplus^{\xi_r}\mathbf{b}_r,\mathbf{b}_t\right)^2+b_h+b_t$
3E-TE-3H-TH	$\mathbf{q}_{(r,E)},\mathbf{e}_r,\mathbf{q}_{(r,H)},\mathbf{b}_r$	E, H	3D in E, H	$-d^{\xi_r}\left(\left(\mathbf{b}_\lambda\otimes\mathbf{q}_{(r,H)}^{\triangleright}\right)\oplus^{\xi_r}\mathbf{b}_r,\mathbf{b}_t\right)^2+b_h+b_t$

Table 2: Six component models and examples of composite models. 3H is a new component model for 3D rotation in hyperbolic space. The composite model 3H-TH performed best in the experiment. E and H in the table represent Euclidean and hyperbolic space, respectively. $\mathbf{q}_r^{\triangleright}$ denotes normalization, $\circ$ denotes Hadamard product, and $\otimes$ denotes Hamilton product. Also, $\mathbf{b}_\gamma := \mathbf{e}_h\circ\mathbf{c}_{(r,E)}+\mathbf{e}_r$ and $\mathbf{b}_\lambda := \mathbf{e}_h\otimes\mathbf{q}_{(r,E)}^{\triangleright}+\mathbf{e}_r$ are used to simplify the formula.

QuatE (Zhang et al., 2019) replaces 2D rotation with a quaternion operation (3D rotation) in quaternion space $\mathbb{Q}$, with the aim of obtaining a more expressive model than RotatE. Given $\mathbf{e}_h\in\mathbb{R}^k,\mathbf{q}_r\in\mathbb{Q}^{\frac{k}{4}},\mathbf{e}_t\in\mathbb{R}^k$, the scoring function of QuatE is

$$s=(\mathbf{e}_h\otimes\mathbf{q}_r^{\triangleright})\cdot\mathbf{e}_t$$

Where $\mathbf{q}_r^{\triangleright}$, $\otimes$, and $\cdot$ represent quaternion normalization, Hamilton product, and dot product, respectively (see Appendix A.1).

3.4 Hyperbolic geometry

We give a brief summary of hyperbolic geometry, and all the hyperbolic geometry equations that we need to use are shown in Appendix A.2, including the logarithmic transformation $\log_{\mathbf{0}}^{\xi_r}(\mathbf{v})$, the exponential transformation $\exp_{\mathbf{0}}^{\xi_r}(\mathbf{y})$, and the Möbius addition ($x\oplus^{\xi_r}y$).

MuRP (Balazevic et al., 2019) is the first paper to introduce translation in hyperbolic space $\mathbb{B}$. Given triple vectors ($\mathbf{b}_h\in\mathbb{B}^k,\mathbf{b}_r\in\mathbb{B}^k,\mathbf{b}_t\in\mathbb{B}^k$), the scoring function is

$$s=-d^{\xi_r}\left(\mathbf{b}_h\oplus^{\xi_r}\mathbf{b}_r,\mathbf{b}_t\right)^2,$$

where $\oplus^{\xi_r}$ and $d^{\xi_r}(.,.)$ represent Möbius addition and hyperbolic distance respectively.

RotH (Chami et al., 2020) aims to replace translation operations with rotation operations in hyperbolic space, similar to how RotatE operates in Euclidean space, in order to capture additional relational patterns. Given triple vectors ($\mathbf{b}_h\in\mathbb{B}^k$, $\mathbf{c}_r\in\mathbb{C}^{\frac{k}{2}},\mathbf{b}_t\in\mathbb{B}^k$), the scoring function is defined as

$$s=-d^{\xi_r}(\mathbf{b}_h\circ\mathbf{c}_r,\mathbf{b}_t)^2,$$

where the elements of $\mathbf{c}_r$ are constrained to be on the unit circle in $\mathbb{C}$.

4 Our Approach

Our proposed model aims to enhance the representation of entities and relations by incorporating various relation patterns, with a particular focus on non-commutative composition, multiplicity, and hierarchy. To achieve this, we leverage techniques such as translation, 3D rotation, and hyperbolic embedding, allowing for a more expressive and comprehensive representation.

4.1 Component models

To maintain a concise representation of the component models for translation and rotation, we have adopted a straightforward naming convention using two letters. The first letter indicates the type of operation: T for translation, 2 for 2D rotation, and 3 for 3D rotation. The second letter indicates the space: E for Euclidean space and H for hyperbolic space. For example, TE represents translation (T) in Euclidean space (E). In total, there are $3\times2=6$ possible combinations of component models that serve as building blocks for creating composite models. The pipeline of any composite model is created by concatenating the component models. Further details regarding various component models and composite models can be found in Table 2.

In the preceding sections, we have introduced TransE (TE), RotatE (2E), QuatE (3E), MuRP (TH), and RotH (2H). Another model not yet proposed is 3H, which does 3D rotation in hyperbolic space. In this study, we propose a new rotation model 3H as follows. Given triple vectors $\mathbf{b}_h \in \mathbb{B}^k, \mathbf{q}_r \in \mathbb{Q}^{\frac{k}{4}}, \mathbf{b}_t \in \mathbb{B}^k$, the scoring function of 3H is

$$s = -d^{\xi_r}(\mathbf{b}_h \otimes \mathbf{q}_r^{\triangleright}, \mathbf{b}_t)^2.$$

4.2 3H-TH model

When examining Table 1, we can observe that 3D rotation is essential for capturing non-commutative properties, while hyperbolic space is crucial for representing hierarchy. Additionally, combining 2d rotation and translation plays an important role in capturing multiplicity; we can expect that the new extension of 3H-TH (3D rotation and translation) possesses similar properties. Taking all these factors into consideration, we will investigate the 3H-TH model that combines these essential elements.

Given head entity $\mathbf{e}_h \in \mathbb{R}^k$ and tail entity $\mathbf{e}_t \in \mathbb{R}^k$, as well as the relation that is split into a 3D rotation part $\mathbf{q}_r \in \mathbb{Q}^{\frac{k}{4}}$ and a translation part $\mathbf{e}_r \in \mathbb{R}^k$, we map entities $\mathbf{e}_h, \mathbf{e}_t$ and the translation relation $\mathbf{e}_r$ from Euclidean space ($\mathbf{e}_h, \mathbf{e}_t, \mathbf{e}_r \in \mathbb{R}^k$) to hyperbolic space ($\mathbf{b}_h, \mathbf{b}_t, \mathbf{b}_r \in \mathbb{B}^k$) using the exponential transformation:

$$\mathbf{b}_\delta = \exp_{\mathbf{0}}^{\xi_r}(\mathbf{e}_\delta) \in \mathbb{B}^k, \delta = h, r, t. \quad (2)$$

as detailed in Equation 9.

The utilization of hyperbolic space in KG models enables the acquisition of hierarchical properties. It is important to note that each relation r in the KG has a unique curvature ξ_r (Chami et al., 2020). Unlike MuRP, where all relations have the same curvature, we train different values of curvature ξ_r for relation r to represent varying degrees of curvature in the hyperbolic space. A higher value of ξ_r for a specific relation signifies a greater degree of hierarchy, resembling a tree-like structure. Conversely, a flatter space represents less hierarchy in the corresponding relation.

The non-commutative property of 3D rotation enables the KG model to perform non-commutative composition, making it more expressive compared to 2D rotation. Therefore, we apply the 3D rotation operation (**3H**) to the mapped head entity in hyperbolic space. Additionally, using rotation and translation operations alone does not allow the model to acquire the multiplicity property. However, combining rotation and translation enables the KG model to exhibit multiplicity. Thus, we utilize Möbius addition ($x \oplus^{\xi_r} y$) as Euclidean translation in hyperbolic space (**TH**). The final operation of 3H-TH model is represented as follows:

$$\mathbf{b}_{(\mathbf{e}_h, \mathbf{e}_r, \mathbf{q}_r)} = (\mathbf{b}_h \otimes \mathbf{q}_r^{\triangleright}) \oplus^{\xi_r} \mathbf{b}_r. \quad (3)$$

Here, $\otimes$ and $\mathbf{q}_r^{\triangleright}$ represent the Hamilton product and normalization, respectively.

4.3 Scoring function and loss

We utilize the hyperbolic distance between the final transformed head entity $\mathbf{b}_{(\mathbf{e}_h, \mathbf{e}_r, \mathbf{q}_r)}$ and the mapped tail entity $\mathbf{b}_t$ as the scoring function:

$$s(h, r, t) = -d^{\xi_r}\left(\mathbf{b}_{(\mathbf{e}_h, \mathbf{e}_r, \mathbf{q}_r)}, \mathbf{b}_t\right)^2 + b_h + b_t. \quad (4)$$

Here, $d^{\xi_r}(.)$ is the hyperbolic distance introduced in Equation 1 with the curvature ξ_r, and $b_v (v \in \mathcal{V})$ represents the entity bias added as a margin in the scoring function (Tifrea et al., 2018; Balazevic et al., 2019). The comparison of various scoring functions, encompassing hyperbolic distance-based, Euclidean distance-based, and dot product-based methods, is detailed in Appendix A.4.1. Moreover, instead of using other negative sampling methods, we uniformly select negative instances for a given triple (h, r, t) by perturbing the tail entity. The model is trained by minimizing the full cross-entropy loss, defined as follows:

$$L = \sum_{t'} \log\left(1 + \exp\left(y_{t'} \cdot s\left(h, r, t'\right)\right)\right) \quad (5)$$

$$y_{t'} = \begin{cases} -1, & \text{if } t' = t \\ 1, & \text{otherwise} \end{cases}$$

4.4 Other composite models

We have introduced a novel component model called 3H, which involves 3D rotation in hyperbolic space. We have also developed a composite model called 3H-TH, which combines 3D rotation and translation in hyperbolic space, as discussed earlier. Furthermore, we have created several other composite models (as shown in Table 2), including 2E-TE (2D Rotation and Translation in Euclidean space), 3E-TE (3D Rotation and Translation in Euclidean space), 2E-TE-2H-TH (2D Rotation and Translation in both Euclidean and Hyperbolic space), and 3E-TE-3H-TH (3D Rotation

Dataset	Entities	Relations	Train	Validation	Test
WN18RR	40,943	11	86,835	3,034	3,134
FB15k-237	14,541	237	272,115	17,535	20,466
FB15K	14,951	1,345	483,142	50,000	59,071

Table 3: Details of the three datasets.

and Translation in both Euclidean and Hyperbolic space).

To examine the effects of integrating translation and rotation, we compare 2E-TE and 3E-TE with their respective counterparts, 2E and 3E. Additionally, we compare 2E-TE-2H-TH and 3E-TE-3H-TH with RotH and 3H-TH to investigate the effects of operations in different spaces. These comparisons allow us to analyze the contributions and implications of different components in the models.

We provide a detailed explanation of 3E-TE-3H-TH because the other models are interpreted as a part of this most complex model. Embeddings of head and tail entities are $\mathbf{e}_h, \mathbf{e}_t \in \mathbb{R}^k$, and embeddings of relation r are $\mathbf{q}_{(r,E)} \in \mathbb{Q}^{\frac{k}{4}}, \mathbf{e}_{(r,E)} \in \mathbb{R}^k, \mathbf{q}_{(r,H)} \in \mathbb{Q}^{\frac{k}{4}}, \mathbf{e}_{(r,H)} \in \mathbb{R}^k$, where $\mathbf{e}_{(r,\alpha)}$ and $\mathbf{q}_{(r,\alpha)}$ are translation and 3D rotation relations, respectively, for space $\alpha \in \{E, H\}$.

We first perform 3D rotation and translation on the head entity in Euclidean space (**3E-TE**) using the following transformation:

$$\mathbf{e}_{\left(\mathbf{e}_h,\mathbf{e}_{(r,E)},\mathbf{q}_{(r,E)}\right)} = \left(\mathbf{e}_h \otimes \mathbf{q}^{\triangleright}_{(r,E)}\right) + \mathbf{e}_{(r,E)} \quad (6)$$

Then we apply the same process as for 3H-TH (Equation 3) to $\mathbf{e}_{\left(\mathbf{e}_h,\mathbf{e}_{(r,E)},\mathbf{q}_{(r,E)}\right)}$, and we use the hyperbolic distance as the scoring function

$$s(h,r,t) = \\ -d^{\xi_r}\left(\left(\mathbf{b}_\lambda \otimes \mathbf{q}^{\triangleright}_{(r,H)}\right) \oplus^{\xi_r} \mathbf{b}_r, \mathbf{b}_t\right)^2 + b_h + b_t. \quad (7)$$

Finally, the loss function is defined by Equation 5 in Section 4.3. We provide more details on several composite models in Table 2.

5 Experiments

We expect that the composite model 3H-TH, which performs both 3D rotation and translation in hyperbolic space, can effectively capture all relation patterns. We aim to validate this expectation through experimentation.

5.1 Experimental setup

Dataset. We evaluate our proposed method on three KG datasets, including WN18RR (Dettmers et al., 2018), FB15K-237 (Toutanova and Chen, 2015), and FB15K (Bordes et al., 2013) with licence CC-BY 2.5. The details of these datasets are shown in Table 3. WN18RR is a subset of WN18 (Dettmers et al., 2018) which is contained in WordNet (Miller, 1995). FK15K is a subset of Freebase (Bollacker et al., 2008), a comprehensive KG including data about common knowledge and FB15K-237 is a subset of FB15K. All three datasets were designed for KGE, and we employ them for KGE tasks, and all three datasets have no individual people or offensive content.

Evaluation metrics. Given a head entity and a relation, we predict the tail entity and rank the correct tail entity against all candidate entities. We use two popular ranking-based metrics: (1) mean reciprocal rank (MRR), which measures the average inverse rank for correct entities: $\frac{1}{n}\sum_{i=1}^{n}\frac{1}{\text{Rank}_i}$. (2) hits on K ($H@K, K \in \{1, 3, 10\}$), which measures the proportion of correct entities appeared in the top K entities.

Baselines. We compare our new model with state-of-the-art (SOTA) methods, namely TransE (Bordes et al., 2013), RotatE (Sun et al., 2019), QuatE (Zhang et al., 2019), MuRP (Balazevic et al., 2019), RotH (Chami et al., 2020), and BiQUE(Guo and Kok, 2021). Alongside these five models and 3H-TH, our comparative models include 3H, 3E-TE, 2E-TE-3H-TH, and 3E-TE-3H-TH. It is worth noting that these comparative models have all been newly developed by us. Significantly, while hyperbolic-based methods indeed require longer training times compared to their Euclidean-based counterparts, it's worth noting that the space and time complexities of all these models remain equivalent. More details of state of the art baselines and discussion refer to Appendix A.7.

Implementation. The key hyperparameters in our implementation include the learning rate, optimizer, negative sample size, and batch size. To determine the optimal hyperparameters, we performed a grid search using the validation data. The optimizer options we considered are Adam (Kingma and Ba, 2014) and Adagrad (Duchi et al., 2011). Finally, we obtain results by selecting the maximum values from three random seeds.

Moreover, to ensure a fair comparison, we incorporated entity bias ($b_v, v \in \mathcal{V}$) into the scoring function for all models (see Table 2). Additionally, we used uniform negative sampling across all

	WN18RR				FB15k-237				FB15K			
Model	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
TransE(TE)	.244	.099	.350	.506	.277	.194	.303	.444	.463	.336	.538	.697
RotatE(2E)	.387	.330	.417	.491	.290	.208	.316	.458	.469	.355	.527	.691
QuatE(3E)	.445	.407	.463	.515	.266	.186	.290	.426	.484	.360	.556	.715
MuRP(TH)	.269	.106	.402	.532	.279	.196	.306	.445	.486	.358	.565	.718
RotH(2H)	.466	.422	.484	.548	.312	.222	.343	.493	.498	.373	.577	.728
BiQUE	.298	.231	.328	.425	.309	.223	.339	.479	-	-	-	-
3H	.467	.429	.486	.541	.277	.195	.302	.444	.500	.375	.576	.726
2E-TE	.448	.421	.474	.522	.262	.184	.283	.419	.494	.373	.568	.725
3E-TE	.456	.408	.467	.518	.261	.184	.282	.414	.496	.376	.572	.725
2E-TE-2H-TH	.469	.428	.487	**.552**	.315	.225	.347	.497	.494	.370	.572	.722
3H-TH	**.473**	**.432**	**.490**	**.552**	**.320**	**.229**	**.351**	**.501**	**.506**	**.383**	**.581**	.731
3E-TE-3H-TH	.469	.424	.481	.546	.316	.227	.346	.499	.504	.379	.580	**.733**

Table 4: Link prediction accuracy results of three datasets in low-dimensional space ($k = 32$). The best score is highlighted in **bold**, and the second-best score is underlined. The **3H-TH** model outperforms other state-of-the-art methods significantly on WN18RR, FB15k-237, and FB15K. Results are statistically significant under paired student's t-test with p-value 0.05 except 2E-TE-2H-TH, more details refer to Appendix A.5

	hierarchy measure								
Relation	Khs_r	$-\xi_r$	TE	2E	2H	BiQUE	2E-TE-2H-TH	**3H-TH**	3E-TE-3H-TH
member meronym	1	-2.9	.407	.304	.390	.245	.407	**.412**	.391
hypernym	1	-2.46	.192	.235	.251	.164	**.271**	.247	.249
has part	1	-1.43	.311	.256	.323	.215	.317	.291	**.337**
instance hypernym	1	-0.82	.492	.488	.488	**.529**	.488	.503	.500
member of domain region	1	-0.78	.442	.442	.462	.423	.423	**.465**	.423
member of domain usage	1	-0.74	.417	.438	.438	**.500**	.438	.441	.417
synset domain topic of	0.99	-0.69	.428	.399	.430	.386	**.434**	.411	.425
also see	0.36	-2.09	.732	.625	.652	.598	.652	.637	.634
derivationally related form	0.07	-3.84	.959	.960	.961	.784	.966	.960	.960
similar to	0.07	-1	1	1	1	.667	1	1	1
verb group	0.07	-0.5	.962	.974	.974	.654	.974	.974	.962

Table 5: Link prediction accuracy results for specific relations sorted by Khs_r. Higher Khs_r or lower $-\xi_r$ indicates a greater degree of hierarchy (Krackhardt, 2014). Accuracy is measured by H@10 in low-dimensional space ($k = 32$) for all 11 relations in WN18RR. The best score is highlighted in **bold**, and the second-best score is underlined. We can observe that the 3H-TH model tends to perform well on relations with larger Khs_r values, indicating its ability to capture hierarchical patterns.

models. We give more details of implementation in Appendix A.3

Finally, we conduct additional experiments to examine the outcomes when we establish equal total parameters (see Appendix A.6).

5.2 Results in low dimensions

Table 4 provides an overview of the overall accuracy in low-dimensional space ($k = 32$). Tables 5 and 6 present detailed results on hierarchy and relation patterns, respectively.

Overall accuracy. Table 4 provides the link prediction accuracy results of WN18RR, FB15K-237, and FB15K in low-dimensional space ($k = 32$). The 3H-TH model outperforms all state-of-the-art models, particularly on the largest dataset FB15K, showcasing the powerful representation capacity achieved by combining 3D rotation and translation in hyperbolic space. Additionally, compared to RotH(2H), the 3H-TH model achieves competitive results across all evaluation metrics, indicating that 3D rotation in hyperbolic space enhances the model's expressiveness. Moreover, the 3H-TH model improves upon previous state-of-the-art Euclidean methods (RotatE and QuatE) by 6.1%, 10.3%, and 10.2% in MRR on WN18RR, FB15K-237, and FB15K, respectively. This comparison highlights the superiority of hyperbolic geometry over Euclidean geometry in low-dimensional KG representation.

Hierarchy. The hierarchy analysis aimed to examine the benefits of using hyperbolic geometry for capturing hierarchy properties. Table 5 presents the H@10 accuracy results for all relations in

Model	Symmetry	Antisymmetry	Composition	Inversion	multiplicity
TransE(TE)	.321	.335	.362	.511	.643
RotatE(2E)	**.454**	**.497**	.338	.512	.663
QuatE(3E)	.324	.388	.357	.541	.683
MuRP(TH)	.335	.359	.361	.542	.666
RotH(2H)	.360	.441	.366	.558	.686
3H	.357	.458	.363	.559	.685
2E-TE	.362	<u>.466</u>	.365	.552	.681
3E-TE	.361	.465	.366	.557	.689
2E-TE-2H-TH	.365	.440	.361	.552	.687
3H-TH	<u>.386</u>	.450	<u>.369</u>	**.566**	**.704**
3E-TE-3H-TH	.361	.444	**.377**	<u>.564</u>	<u>.691</u>

Table 6: Link prediction accuracy for specific relation patterns. Accuracy is measured by MRR for FB15K in low-dimensional space ($k = 32$). **Bold** indicates the best score, and <u>underline</u> represents the second-best score. The 3H-TH model achieves the best or second-best performance on the *symmetry, composition, inversion,* and *multiplicity* properties.

WN18RR, sorted by Khs_r, the Krackhardt hierarchy score (Krackhardt, 2014) and ξ_r, estimated graph curvature (Chami et al., 2020). A higher Khs_r or lower $-\xi_r$ indicates a higher degree of hierarchy in the relations. The table confirms that the first 7 relations exhibit hierarchy, while the remaining relations do not. From the results, we observe that although Euclidean embeddings (TransE, RotatE) and hyperbolic embeddings (RotH, 3H-TH) perform similarly on non-hierarchical relations like *verb group* and *similar to*, hyperbolic embeddings outperform significantly on top 7 hierarchical relations. More discussion of this part refers to Appendix A.4.2

Relation Patterns. The relation patterns analysis aimed to assess the performance of different models on specific relation patterns. Table 6 presents the MRR accuracy results for relation patterns in FB15K, measured in low-dimensional space ($k = 32$). We can observe that the 3H-TH model outperforms on relation patterns such as *symmetry*, *composition*, *inversion*, and *multiplicity*, either achieving the best score or the second-best score. Additionally, the accuracy of 3H-TH on *antisymmetry* is also very high. (For a more comprehensive analysis of the results for various composite models and the frequency distribution of various relation patterns within the datasets, please consult the Appendix A.4.3 and A.4.4 respectively)

5.3 Results in high dimensions

Table 7 displays the link prediction accuracy results for WN18RR in high-dimensional space ($k = 200$). As anticipated, the 3H-TH model and some other composite models (2E-TE-2H-TH, 3E-TE-3H-TH) achieve new state-of-the-art (SOTA) results. However, the accuracy is comparable to that of RotH and Euclidean space methods. This indicates that Euclidean and hyperbolic embeddings perform similarly when the embedding dimension is large. For additional experiments and results in higher dimensions, refer to Appendix A.4.5.

Model	MRR	H@1	H@3	H@10
TransE(TE)	.263	.107	.380	.532
RotatE(2E)	.396	.384	.399	.419
QuatE(3E)	.487	.442	.503	.573
MuRP(TH)	.265	.105	.392	.531
RotH(2H)	.490	.444	.507	.578
BiQUE	<u>.491</u>	.451	.508	.566
3H	.484	.440	.500	.571
2E-TE	.393	.382	.396	.415
3E-TE	.490	.445	.506	.578
2E-TE-2H-TH	**.493**	.446	<u>.509</u>	<u>.585</u>
3H-TH	**.493**	<u>.447</u>	<u>.509</u>	**.587**
3E-TE-3H-TH	**.493**	**.448**	**.510**	.579

Table 7: Link prediction accuracy results for WN18RR in high-dimensional space ($k = 200$).

6 Conclusion

In this study, we propose the 3H-TH model for KGE to address multiple relation patterns, including *symmetry, antisymmetry, inversion, commutative composition, non-commutative composition, hierarchy,* and *multiplicity.* By combining 3D rotation and translation in hyperbolic space, the model effectively represents entities and relations. Experimental results demonstrate that the 3H-TH model achieves excellent performance in low-dimensional space. Moreover, the performance difference becomes smaller in high-dimensional space, although the model still performs well.

Limitations

Limited improvements in high dimensions While our approach 3H-TH shows substantial improvement over baseline models in a low-dimensional ($k = 32$) KGE setting, we observe that as we move towards higher dimensions ($k = 200, 300, 500$), our techniques tend to converge and exhibit similar results to Euclidean base models. As an illustration, the link prediction accuracy of the 3H-TH model is similar to the Euclidean space methods, as evidenced in Table 7 and some results provided in Appendix A.4. The difference in representational capacity between geometric spaces (Euclidean and hyperbolic space) becomes quite pronounced in lower dimensions. However, this gap may lessen or even disappear as the dimension is increased.

Rotation in hyperbolic space Examining strictly from mathematical and geometric perspectives, it is correct to perform translations in hyperbolic space. However, conducting rotational operations (2D and 3D rotation) in hyperbolic space akin to those in Euclidean space lacks a certain level of rigor.

Acknowledgements

References

Ivana Balazevic, Carl Allen, and Timothy Hospedales. 2019. Multi-relational poincaré graph embeddings. *Advances in Neural Information Processing Systems*, 32.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250.

Silvere Bonnabel. 2013. Stochastic gradient descent on riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9):2217–2229.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26.

Zongsheng Cao, Qianqian Xu, Zhiyong Yang, Xiaochun Cao, and Qingming Huang. 2021. Dual quaternion knowledge graph embeddings. *Proceedings of the AAAI conference on artificial intelligence*, 35(8):6894–6902.

Ines Chami, Adva Wolf, Da-Cheng Juan, Frederic Sala, Sujith Ravi, and Christopher Ré. 2020. Low-dimensional hyperbolic knowledge graph embeddings. *arXiv preprint arXiv:2005.00545*.

Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. 2019. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems*, 32.

Sanxing Chen, Xiaodong Liu, Jianfeng Gao, Jian Jiao, Ruofei Zhang, and Yangfeng Ji. 2020. Hitter: Hierarchical transformers for knowledge graph embeddings. *arXiv preprint arXiv:2008.12813*.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. *Proceedings of the AAAI conference on artificial intelligence*, 32(1).

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7).

Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. 2018. Hyperbolic neural networks. *Advances in neural information processing systems*, 31.

Jia Guo and Stanley Kok. 2021. Bique: Biquaternionic embeddings of knowledge graphs. *arXiv preprint arXiv:2109.14401*.

Chi Han, Qizheng He, Charles Yu, Xinya Du, Hanghang Tong, and Heng Ji. 2023. Logical entity representation in knowledge-graphs for differentiable rule learning. *arXiv preprint arXiv:2305.12738*.

Yanchao Hao, Yuanzhe Zhang, Kang Liu, Shizhu He, Zhanyi Liu, Hua Wu, and Jun Zhao. 2017. An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 221–231.

Jiabang He, Liu Jia, Lei Wang, Xiyao Li, and Xing Xu. 2023. Mocosa: Momentum contrast for knowledge graph completion with structure-augmented pre-trained language models. *arXiv preprint arXiv:2308.08204*.

Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Knowledge graph embedding via dynamic mapping matrix. In *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (volume 1: Long papers)*, pages 687–696.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

David Krackhardt. 2014. Graph theoretical dimensions of informal organizations. In *Computational organization theory*, pages 107–130. Psychology Press.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. *Proceedings of the AAAI conference on artificial intelligence*, 29(1).

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

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

Afshin Sadeghi, Hirra Abdul Malik, Diego Collarana, and Jens Lehmann. 2021. Relational pattern benchmarking on the knowledge graph link prediction task. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, pages 697–706.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint arXiv:1902.10197*.

Alexandru Tifrea, Gary Bécigneul, and Octavian-Eugen Ganea. 2018. Poincar\'e glove: Hyperbolic word embeddings. *arXiv preprint arXiv:1810.06546*.

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 *International conference on machine learning*, pages 2071–2080. PMLR.

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

Liang Wang, Wei Zhao, Zhuoyu Wei, and Jingming Liu. 2022a. Simkgc: Simple contrastive knowledge graph completion with pre-trained language models. *arXiv preprint arXiv:2203.02167*.

Xintao Wang, Qianyu He, Jiaqing Liang, and Yanghua Xiao. 2022b. Language models as knowledge embeddings. *arXiv preprint arXiv:2206.12617*.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. *Proceedings of the AAAI conference on artificial intelligence*, 28(1).

Chenyan Xiong, Russell Power, and Jamie Callan. 2017. Explicit semantic ranking for academic search via knowledge graph embedding. In *Proceedings of the 26th international conference on world wide web*, pages 1271–1279.

Bishan Yang and Tom Mitchell. 2019. Leveraging knowledge bases in lstms for improving machine reading. *arXiv preprint arXiv:1902.09091*.

Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. 2016. Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 353–362.

Ningyu Zhang, Xin Xie, Xiang Chen, Yongheng Wang, Xu Cheng, and Huajun Chen. 2022. Reasoning through memorization: Nearest neighbor knowledge graph embeddings. *arXiv preprint arXiv:2201.05575*.

Shuai Zhang, Yi Tay, Lina Yao, and Qi Liu. 2019. Quaternion knowledge graph embeddings. *Advances in neural information processing systems*, 32.

A Appendix

A.1 Hamilton's quaternions

A quaternion q is composed of one real number component and three imaginary number components. It can be represented as $\mathrm{q} = a + b\mathrm{i} + c\mathrm{j} + d\mathrm{k}$, where a, b, c, and d are real numbers, and i, j, and k are imaginary numbers. The real part is represented by a, while the imaginary parts are represented by $b\mathrm{i}$, $c\mathrm{j}$, and $d\mathrm{k}$.

Hamilton's rules govern quaternion algebra and include the following: (1). $\mathrm{i}^2 = \mathrm{j}^2 = \mathrm{k}^2 = \mathrm{ijk} = -1$, (2). $\mathrm{ij} = \mathrm{k}, \mathrm{ji} = -\mathrm{k}, \mathrm{jk} = \mathrm{i}, \mathrm{kj} = -\mathrm{i}, \mathrm{ki} = \mathrm{j}, \mathrm{ik} = -\mathrm{j}$

In addition to these rules, various mathematical operations can be performed with quaternions:

Normalization. When real elements of quaternion are numbers, $\mathrm{q}^{\triangleright} = \frac{\mathrm{q}}{|\mathrm{q}|} = \frac{a+b\mathrm{i}+c\mathrm{j}+d\mathrm{k}}{\sqrt{a^2+b^2+c^2+d^2}}$. On the other hand, when the real elements of a quaternion, denoted as $\mathbf{q}_r$, are represented by vectors, the normalization formula needs to be modified. In this case, the quaternion normalization $\mathbf{q}_r^{\triangleright}$ is given by:

$$\mathbf{q}_r^{\triangleright} = \frac{\mathbf{q}_r}{|\mathbf{q}_r|} = \frac{\mathbf{a} + \mathbf{b}\mathrm{i} + \mathbf{c}\mathrm{j} + \mathbf{d}\mathrm{k}}{\sqrt{\mathbf{a}^\top\mathbf{a} + \mathbf{b}^\top\mathbf{b} + \mathbf{c}^\top\mathbf{c} + \mathbf{d}^\top\mathbf{d}}}$$

Here, $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, and $\mathbf{d}$ represent vector representations of the real components, and $\mathbf{a}^\top$, $\mathbf{b}^\top$, $\mathbf{c}^\top$, and $\mathbf{d}^\top$ denote the transpose of the respective vectors. The numerator consists of the vector components, and the denominator involves the Euclidean norm of the vector elements.

Dot product. Given $\mathrm{q}_1 = a_1 + b_1\mathrm{i} + c_1\mathrm{j} + d_1\mathrm{k}$ and $\mathrm{q}_2 = a_2 + b_2\mathrm{i} + c_2\mathrm{j} + d_2\mathrm{k}$, we can obtain the dot product of q_1 and q_2:

$$\mathrm{q}_1 \cdot \mathrm{q}_2 = a_1a_2 + b_1b_2 + c_1c_2 + d_1d_2.$$

Hamilton product. The multiplication of two quaternions follows from the basic Hamilton's rule. Given q_1 and q_2, the multiplication is:

$$\begin{aligned} \mathrm{q}_1 \otimes \mathrm{q}_2 = &(a_1a_2 - b_1b_2 - c_1c_2 - d_1d_2) \\ &+ (a_1b_2 + b_1a_2 + c_1d_2 - d_1c_2)\mathrm{i} \\ &+ (a_1c_2 - b_1d_2 + c_1a_2 + d_1b_2)\mathrm{j} \\ &+ (a_1d_2 + b_1c_2 - c_1b_2 + d_1a_2)\mathrm{k} \end{aligned} \tag{8}$$

Equation (8) presents Hamilton's product as non-commutative, which shows that 3D rotation can enable the model to perform non-commutative.

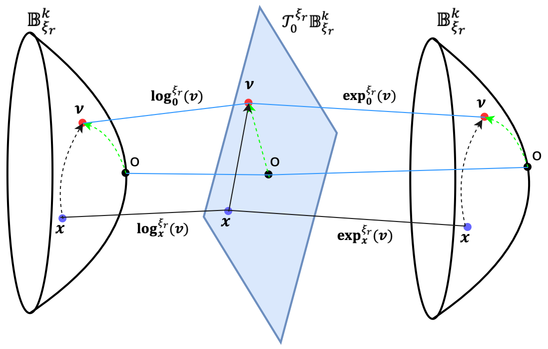

Figure 2: The logarithmic transformation $\log_{\mathbf{0}}^{\xi_r}(\mathbf{v})$ ($\mathbb{B}^k_{\xi_r} \rightarrow \mathcal{T}_{\mathbf{0}}^{\xi_r}\mathbb{B}^k_{\xi_r}$) and the exponential transformation $\exp_{\mathbf{0}}^{\xi_r}(\mathbf{v})$ ($\mathcal{T}_{\mathbf{0}}^{\xi_r}\mathbb{B}^k_{\xi_r} \rightarrow \mathbb{B}^k_{\xi_r}$)

A.2 Hyperbolic geometry

Hyperbolic geometry, characterized by continuous negative curvature, is a non-Euclidean geometry. One way to represent hyperbolic space is through the k-dimensional Poincaré ball model with negative curvature $-\xi_r$ ($\xi_r > 0$). In this model, hyperbolic space is expressed as $\mathbb{B}^k_{\xi_r} = \{\mathbf{x} \in \mathbb{R}^k : \|\mathbf{x}\|^2 < \frac{1}{\xi_r}\}$, where $\|\cdot\|$ denotes the L2 norm. The Poincaré ball model provides a geometric framework to understand and study hyperbolic geometry.

In the Poincaré ball model, for any point $\mathbf{x} \in \mathbb{B}^k_{\xi_r}$, all possible directions of paths are contained within the tangent space $\mathcal{T}_{\mathbf{x}}^{\xi_r}$, which is a k-dimensional vector space. The tangent space connects Euclidean and hyperbolic space, meaning that $\mathcal{T}_{\mathbf{x}}^{\xi_r}\mathbb{B}^k_{\xi_r} = \mathbb{R}^k$. Since the tangent space exhibits Euclidean geometric properties, vector addition and multiplication can be performed in this space just like in Euclidean space.

Moreover, the logarithmic transformation $\log_{\mathbf{0}}^{\xi_r}(\mathbf{v})$ maps a point in the Poincaré ball $\mathbb{B}^k_{\xi_r}$ to the tangent space $\mathcal{T}_{\mathbf{0}}^{\xi_r}\mathbb{B}^k_{\xi_r}$. Specifically, it maps a point from the origin in the direction of a vector $\mathbf{v}$. Conversely, the exponential transformation $\exp_{\mathbf{0}}^{\xi_r}(\mathbf{y})$ performs the reverse mapping. It maps a point from the tangent space $\mathcal{T}_{\mathbf{0}}^{\xi_r}\mathbb{B}^k_{\xi_r}$ back to the Poincaré ball, originating from the origin in the direction of a vector $\mathbf{y}$ (see Fig. 2). These transformations facilitate the conversion between the Poincaré ball and its associated tangent space, enabling geometric operations in both spaces (Chami et al., 2020).

$$\exp_{\mathbf{0}}^{\xi_r}(\mathbf{v}) = \tanh(\sqrt{\xi_r}||\mathbf{v}||)\frac{\mathbf{v}}{\sqrt{\xi_r}||\mathbf{v}||}, \tag{9}$$

$$\log_{\mathbf{0}}^{\xi_r}(\mathbf{y}) = \tanh^{-1}(\sqrt{\xi_r}||\mathbf{y}||)\frac{\mathbf{y}}{\sqrt{\xi_r}||\mathbf{y}||}. \tag{10}$$

We introduce the logarithmic transformation $\log_{\mathbf{0}}^{\xi_r}(\mathbf{v})$ ($\mathbb{B}^k_{\xi_r} \to \mathcal{T}_{\mathbf{0}}^{\xi_r}\mathbb{B}^k_{\xi_r}$) and exponential transformation $\exp_{\mathbf{0}}^{\xi_r}(\mathbf{y})$ ($\mathcal{T}_{\mathbf{0}}^{\xi_r}\mathbb{B}^k_{\xi_r} \to \mathbb{B}^k_{\xi_r}$) from the origin in the direction of a vector. Generally, the logarithmic transformation $\log_{\mathbf{x}}^{\xi_r}(\mathbf{v})$ ($\mathbb{B}^k_{\xi_r} \to \mathcal{T}_{\mathbf{x}}^{\xi_r}\mathbb{B}^k_{\xi_r}$) and exponential transformation $\exp_{\mathbf{x}}^{\xi_r}(\mathbf{y})$ ($\mathcal{T}_{\mathbf{x}}^{\xi_r}\mathbb{B}^k_{\xi_r} \to \mathbb{B}^k_{\xi_r}$) from $\mathbf{x}$ in the direction of a vector $\mathbf{y}, \mathbf{v}$ respectively (Balazevic et al., 2019) are:

$$\log_{\mathbf{x}}^{\xi_r}(\mathbf{y}) = \frac{2}{\sqrt{\xi_r}\lambda_{\mathbf{x}}^{\xi_r}}\tanh^{-1}\left(\sqrt{\xi_r}\left\|-\mathbf{x}\oplus^{\xi_r}\mathbf{y}\right\|\right)\frac{-\mathbf{x}\oplus^{\xi_r}\mathbf{y}}{\|-\mathbf{x}\oplus^{\xi_r}\mathbf{y}\|},$$

$$\exp_{\mathbf{x}}^{\xi_r}(\mathbf{v}) = \mathbf{x}\oplus^{\xi_r}\left(\tanh\left(\sqrt{\xi_r}\frac{\lambda_{\mathbf{x}}^{\xi_r}\|\mathbf{v}\|}{2}\right)\frac{\mathbf{v}}{\sqrt{\xi_r}\|\mathbf{v}\|}\right).$$

Besides, we apply Möbius addition ($\mathbf{x} \oplus^{\xi_r} \mathbf{y}$) (Ganea et al., 2018) to replace Euclidean translation in hyperbolic space, considering that the hyperbolic space can be regarded as a roughly vectorial structure (Ungar, 2008):

$$\mathbf{x}\oplus^{\xi_r}\mathbf{y} = \frac{(1+2\xi_r\mathbf{x}^T\mathbf{y}+\xi_r\|\mathbf{y}\|^2)\mathbf{x}+(1-\xi_r\|\mathbf{x}\|^2)\mathbf{y}}{1+2\xi_r\mathbf{x}^T\mathbf{y}+{\xi_r}^2\|\mathbf{x}\|^2\|\mathbf{y}\|^2} \tag{11}$$

A.3 More details about Implementation

In previous work, MuRP employed Riemannian Stochastic Gradient Descent (RSGD) (Bonnabel, 2013), which is typically required for optimization in hyperbolic space. However, RSGD is difficult to use in real applications. Since it has been demonstrated that tangent space optimization is effective (Chami et al., 2019), we first define all the 3H-TH parameters in the tangent space at the origin and apply conventional Euclidean methods to optimize the embeddings. Afterward, we use exponential transformation to map the parameters from Euclidean space to hyperbolic space. Therefore, all the 3H-TH model parameters $\left\{(\mathbf{e}_r, \mathbf{q}_r, \xi_r)_{r\in\mathcal{R}}, (\mathbf{e}_v, b_v)_{v\in\mathcal{V}}\right\}$ are now Euclidean parameters that can be learned using conventional Euclidean optimization methods such as Adam or Adagrad.

Furthermore, models are trained on a single RTX8000 (48GB) GPU. For 3H-TH and related composite models, training times are approximately 1 hour for WN18RR, 4 hours for FB15K-237, and 10 hours for FB15K. We use PyTorch and Numpy as the additional tools to conduct our experiment. We use ChatGPT in our paper writing.

A.4 Additional experiments and results

We have included supplementary experiments in the appendix to validate our methods. A.4.1 focuses on comparing various scoring functions, providing additional experiments and results that demonstrate the superiority of hyperbolic-distance-based scoring functions over others. A.4.2 utilizes statistical analyses of each relation to elucidate why TransE excels in specific hierarchy relations. In the A.4.3, we provide additional explanations and results related to relation patterns, offering an analysis of the roles and effects of various composite models. A.4.4 presents the frequency distribution of various relation patterns, shedding light on the importance of each pattern. Lastly, A.4.5 presents additional results in high-dimensional space, encompassing accuracy and hierarchy results.

A.4.1 Comparison of various scoring function

In our 3H-TH model, we employed a distance-based scoring function (hyperbolic distance) to replace the inner-product to better utilize the advantages of the hyperbolic space, particularly its ability to better capture hierarchical properties. However, distance-based scoring function may lose the Complex Relation properties (1-1, 1-n, n-1, n-n) compared with dot product scoring function which utilized by QuatE(Zhang et al., 2019). Therefore, we conduct supplementary experiments to verify which scoring function is best.

We introduce three additional models for comparison alongside the 3H-TH model. The first model, denoted as 3H-TH (Project & Inner product), entails transforming the head entity from hyperbolic space to Euclidean space within the 3H-TH model, utilizing the inner product as its scoring function. The second model, referred to as QuatE (Inner product), corresponds to the original QuatE model employing the dot product as its scor-

Model	MRR	H@1	H@3	H@10	1-1 (1.34%)	1-n (15.16%)	n-1 (47.45%)	n-n (36.06%)
3H-TH (Hyperbolic distance)	.473	.435	.485	.547	.911	.226	.190	.931
3H-TH (Project & Inner product)	.356	.342	.362	.380	.703	.057	.029	.900
QuatE (Inner product)	.358	.264	.413	.529	.921	.085	.054	.902
QuatE (Euclidean distance)	.445	.407	.463	.515	.889	.176	.164	.899

Table 8: The accuracy results (MRR, H@1,3,10) and complex relation MRR results (1-1, 1-n, n-1, n-n) of various scoring function methods in WN18RR.

Model	MRR	H@1	H@3	H@10	1-1 (1.34%)	1-n (15.16%)	n-1 (47.45%)	n-n (36.06%)
3H-TH (Hyperbolic distance)	.507	.387	.577	.728	.601	.524	.528	.494
3H-TH (Project & Inner product)	.500	.385	.564	.721	.535	.497	.516	.497
QuatE (Inner product)	.457	.345	.514	.675	.400	.450	.485	.454
QuatE (Euclidean distance)	.484	.360	.556	.715	.578	.504	.515	.480

Table 9: The accuracy results ((MRR, H@1,3,10)) and complex relation MRR results (1-1, 1-n, n-1, n-n) of various scoring function methods in FB15K.

Relation	Num-relations(Percentage)
member meronym	253 (0.87%)
hypernym	1251 (39.92%)
has part	172 (5.49%)
instance hypernym	122 (3.89%)
member of domain region	26 (0.83%)
member of domain usage	24 (0.77%)
synset domain topic of	114 (3.64%)
also see	56 (1.79%)
derivationally related form	1074 (34.27%)
similar to	3 (0.09%)
verb group	39 (1.24%)

Table 10: Frequency distribution of different relations in WN18RR.

ing function. The final model, QuatE (Euclidean distance), employs Euclidean distance as the scoring function within the QuatE model. In Table 8 and 9, we present the overall mean reciprocal rank (MRR), overall accuracies (H@1,3,10), and MRR specifically for complex relation patterns (1-1, 1-n, n-1, n-n) in the WN18RR and FB15K datasets, respectively. The values in parentheses denote the percentages of triple instances. These experiments were conducted in a low-dimensional space (dim = 32).

Across both datasets, the 3H-TH model using hyperbolic distance consistently offers better performance than other models. Which suggesting that a hyperbolic distance-based scoring function can better utilize the strengths of hyperbolic space. Besides, when contrasting 3H-TH (Hyperbolic distance) and 3H-TH (Project & Inner product) across both datasets, the former consistently shows better results in terms of accuracy and complex relation metrics. Finally, the performance of QuatE (Euclidean distance) surpasses QuatE (Inner product) in both datasets in low-dimensional space. This implies that, particularly in low-dimensional spaces, distance-based methods can provide a more precise measure of the differences between two vectors than inner-product based methods. In conclusion, the distance-based scoring function performs BETTER than the inner-product one in QuatE, especially in low dimensions, while they perform similarly in high dimensions. Our proposed 3H-TH uses distance in hyperbolic space and performs even better than QuatE.

A.4.2 Explanation of TransE performs well on certain hierarchy relations

Phenomena have been observed where TransE (TE) exhibits noteworthy performance on specific hierarchy relations, as exemplified in Table 5. Notably, the results of relations such as *member meronym*, *member of domain region*, and *member of domain usage* indicate that TransE can achieve high accuracy, even though they cannot perform better than 3H-TH. This phenomenon can be attributed to the unbalanced distribution of individual relations within the WN18RR dataset, as demonstrated in Table 10.

As can be seen from the table, TransE methods, which perform well, such as *member meronym* (8.07%), *member of domain region* (0.83%), and *member of domain uasage* (0.77%), have a relatively low proportion in the overall test set. This can introduce an element of randomness to the results. However, in relation with a higher proportion like *hypernym* (39.92%), the performance of TransE is considerably inferior to hyperbolic methods (3H-TH, etc.).

A.4.3 Additional relation pattern results explanation

To the best of our knowledge, no previous work in the KGE domain presents detailed results for these relation patterns, although several methods provide visualization results like (Sun et al., 2019) or theoretical explanations for multiple patterns like (Cao et al., 2021). We obtain the FB15K test data for *symmetry*, *antisymmetry*, *inversion*, and *composition* from (Sadeghi et al., 2021), meanwhile, we use multiple pattern properties to classify them from the FB15K test data. The MRR results of relation patterns on FB15K in low-dimensional space (dim = 32), including *symmetry*, *antisymmetry*, *inversion*, *composition*, and *multiple*, are summarized in Table 6.

In addition to the MRR accuracy outcomes detailed in the primary document, we also have supplementary findings and results to demonstrate. RotatE performs better on Symmetry and Antisymmetry because this model is simple and targeted to these two properties. Moreover, 3D rotation-based methods (3H-TH, 3E-TE-3H-TH) tend to perform better than 2D rotation-based methods (RotH, 2E-TE-2H-TH) on composition patterns in Hyperbolic space, which may indicate that 3D rotation can help the model to acquire non-commutative property on the composition pattern, although we did not classify the test data to test this. Finally, for evaluating multiple patterns, we obverse that 3H-TH can achieve the best results and combination-based methods (combine translation and rotation)(2E-TE, 3E-TE) perform better than the single-based methods (TransE, RotatE, QuatE) on the multiple patterns, which shows that combination-based methods enable model powerful representation capability of multiple patterns.

A.4.4 Frequency distribution of various relation patterns

A pivotal aspect of our research focuses on concurrently solving various relation patterns. Consequently, it becomes imperative to delve into the statistical analysis of the frequency distribution associated with these various relation patterns within the datasets, as well as engage in a comprehensive discourse on the significance attributed to these relation patterns. In this context, we present an overview of the available data and employ specialized algorithms to calculate the frequencies of specific relation patterns embedded within the WN18RR, FB15K-237, and FB15K datasets.

Triple	Symmetry	Antisymmetry	Inversion
Train(483142)	20333(4.2%)	63949(13.2%)	66385(13.7%)
Valid(50000)	3392(6.78%)	25396(50.79%)	8798(17.60%)
Test(59071)	3375(5.71%)	26020(44.05%)	8798(14.89%)

Table 11: Frequency and proportion of (anti)symmetry and inversion in FB15K.

(Anti)symmetry, Inversion, Composition Only (Anti)symmetry, Inversion, Composition were discovered and studied before RotatE (Sun et al., 2019), which provided some dataset details in their paper. In their seminal work, they elucidated that the WN18RR and FB15K237 datasets primarily encompass the *symmetry*, *antisymmetry*, and *composition* relation patterns, whereas the FB15K dataset predominantly comprises the *symmetry*, *antisymmetry*, and *inversion* relation patterns. Furthermore, Sadeghi et al. (Sadeghi et al., 2021) have conducted a detailed analysis of the frequency distribution of *(anti)symmetry* and *inversion* relation patterns within the FB15K dataset, which is presented in Table 11.

From the aforementioned literature and data, it is evident that the proportion of the four relation patterns: *Symmetry*, *Antisymmetry*, *Inversion*, and *Composition*, is substantial. This underscores their research significance and value.

Hierarchy Given that the hierarchy is a tree-like structure, it's challenging to provide a quantitative statistical result. Therefore, we select and compare the quantity and percentage of the top 7 more hierarchical relations in Table 5 from the WN18RR dataset, the training set has 86,835 triples, with 62.9% (54,603) being hierarchy relations. The test set contains 3,134 triples, 62.6% (1,962) of which are hierarchy relations, while the validation set includes 3,034 triples, 61.6% (1,869) of them being hierarchy relations. Based on the statistical results from WN18RR, the proportion of *hierarchy* relations remains substantial.

Multiplicity The extraction of this relation pattern is based on the properties of multiplicity, and we derived it from the dataset using the corresponding algorithm. Subsequently, we carried out statistics related to multiplicity on various datasets which has been shown in Table 14.

From the statistical results in the Table 14, it can be observed that on smaller datasets like WN18RR, where the number of relations is limited (number = 11), the proportion of Multiplicity relations is relatively low. However, its proportion is still sig-

Model	MRR	H@1	H@3	H@10
TransE(TE)	.262	.108	.379	.531
RotatE(2E)	.387	.377	.390	.406
QuatE(3E)	.490	.444	.506	.580
MuRP(TH)	.263	.102	.388	.529
RotH(2H)	.488	.443	.506	.575
3H	.491	**.447**	.507	.576
2E-TE	.390	.379	.395	.411
3E-TE	**.492**	.444	**.511**	.581
2E-TE-2H-TH	.490	.446	.505	.578
3H-TH	.491	.443	**.511**	.581
3E-TE-3H-TH	**.492**	.446	.508	**.582**

Table 12: The link prediction accuracy results of WN18RR in high-dimensional space ($k = 300$).

Model	MRR	H@1	H@3	H@10
TransE(TE)	.260	.104	.378	.532
RotatE(2E)	.380	.372	.383	.395
QuatE(3E)	.490	.443	.507	**.580**
MuRP(TH)	.260	.102	.380	.529
RotH(2H)	.489	.443	.508	.579
3H	.487	.441	.503	.575
2E-TE	.383	.372	.388	.400
3E-TE	**.492**	**.445**	.509	.585
2E-TE-2H-TH	.489	.442	.507	.579
3H-TH	.491	**.445**	**.510**	**.580**
3E-TE-3H-TH	.487	.443	.502	.578

Table 13: The link prediction accuracy results of WN18RR in high-dimensional space ($k = 500$).

nificant in larger datasets like FB15K and FB15K-237, especially in the larger training sets. Thus, the Multiplicity relation patterns are also crucial and hold research significance.

A.4.5 Additional high-dimensional results

Table 12 and Table 13 display the link prediction accuracy results of WN18RR in high-dimensional space with dimensions $k = 300$ and $k = 500$ respectively. These tables demonstrate that the 3H-TH model, as well as other composite models, achieve state-of-the-art (SOTA) performance, comparable to RotH and Euclidean space methods. This suggests that in large embedding dimensions, both Euclidean and hyperbolic embeddings exhibit similar performance.

Furthermore, Table 15 presents the H@10 results for each relation in WN18RR using high-dimensional embeddings. In comparison to Euclidean embedding methods (TransE, RotatE), hyperbolic embedding methods (RotH, 3H-TH, 3E-TE-3H-TH) perform better on hierarchical relations

Dataset	Num-triples	Multiplicity
WN18RR(Train)	86835	218(0.25%)
WN18RR(Valid)	3034	0(0.00%)
WN18RR(Test)	3134	0(0.00%)
FB15K-237(Train)	272113	49214(18.09%)
FB15K-237(Valid)	17535	160(0.91%)
FB15K-237(Test)	20466	224(1.09%)
FB15K(Train)	483142	152194(31.50%)
FB15K(Valid)	50000	2461(4.92%)
FB15K(Test)	59071	3341(5.66%)

Table 14: Frequency and proportion of Multiplicity in WN18RR, FB15K-237, and FB15K.

such as *member meronym*, *hypernym*, and *has part*. This indicates that hyperbolic embeddings can effectively capture and model hierarchy even in high-dimensional spaces.

A.5 Statistical significance test

We use the WN18RR dataset for experimentation in low-dimensional space (dim = 32), the details of which can be found in Table 4 of the paper. And we use the MRR of each triple in 3H-TH as x, and the MRR of each triple in the other models (RotH, 3H, 2E-TE, 3E-TE, 2E-TE-2H-TH, 3E-TE-3H-TH) as y. Then, we calculated the standard deviation (Std(x-y)), variance (Var(x-y)), standard error (Se(x-y)) of the differences (x-y), and paired student's t-test (P-value2) (The test Samples are 3134, the degree of freedom is 3133, which guarantees that appropriateness of using t-test). The detailed experimental results are shown in the Table 16.

From the paired student's t-test results, the normal approximation (dpvalue1) is almost identical since the test sample (3134) is large. When comparing MRR and its p-value2, all the model are worse than 3H-TH. The difference are significant ($p < 0.05$) except for 2E-TE-2H-TH ($p = 0.075$). For the past model RotH ($p = 0.0024 < 0.01$), we can claim that RotH is significantly worse than 3H-TH. As for 2E-TE-2H-TH ($p > 0.05$), this model represents a novel approach that has not been proposed previously. Based on the p-value, we can assert the significant value of this model.

A.6 Additional composite model experiments

The TE model has a single relation representation, denoted as $\mathbf{e}_r$. On the other hand, the 3E-TE-3H-TH model has four relation embeddings, namely $\mathbf{q}_{(r,E)}, \mathbf{e}_r, \mathbf{q}_{(r,H)}, \mathbf{b}_r$. Consequently, the total parameters for each model differ when we set the en-

	hierarchy measure							
Relation	Khs_r	$-\xi_r$	TE	2E	2H	BiQUE	**3H-TH**	3E-TE-3H-TH
member meronym	1	-2.9	.413	.393	**.431**	.378	.421	.427
hypernym	1	-2.46	.210	.309	**.310**	.289	.304	.303
has part	1	-1.43	.320	.323	.355	.351	**.384**	.346
instance hypernym	1	-0.82	.500	.533	.537	**.586**	.533	.504
member of domain region	1	-0.78	.423	.423	.481	.481	.464	**.500**
member of domain usage	1	-0.74	.438	.458	.458	**.479**	.458	.458
synset domain topic of	0.99	-0.69	.461	.513	.509	**.540**	.522	.522
also see	0.36	-2.09	.741	.652	.661	.723	.679	.679
derivationally related form	0.07	-3.84	.956	.969	.969	.966	.966	.966
similar to	0.07	-1	1	1	1	1	1	1
verb group	0.07	-0.5	.936	.974	.974	.974	.974	.974

Table 15: Comparison of H@10 for WN18RR relations in high-dimensional space ($k = 200$). **Bold** indicates the best score, and underline represents the second-best score.

Model	MRR	Std(x-y)	Var(x-y)	Se(x-y)	P-value2
3H-TH	.473	-	-	-	-
RotH(2H)	.466	.122	.015	.002	2.36e-03
3H	.467	.128	.017	.002	1.18e-05
2E-TE	.448	.135	.018	.002	1.14e-24
3E-TE	.456	.123	.015	.002	4.44e-15
2E-TE-2H-TH	.469	.122	.015	.002	7.54e-02
3E-TE-3H-TH	.469	.125	.016	.002	4.21e-02

Table 16: Statistical significance test for 3H-TH and other baseline models in WN18RR dataset.

tity dimensions k to the same value. Alternatively, we conduct additional experiments to examine the outcomes when we establish equal total parameters, encompassing both entity and relation parameters. This comparison takes into account the degrees of freedom associated with each relation type. Specifically, the translation relation $\mathbf{e}_r$ has k parameters in each relation, the 2D rotation relation $\mathbf{c}_r$ has $\frac{1}{2}k$ parameters in each relation with the constraint $|(\mathbf{c}_r)_i| = 1$, and the 3D rotation relation $\mathbf{q}_r$ has $\frac{3}{4}k$ parameters in each relation with the normalization constraint $\mathbf{q}_r^{\triangleright}$). For more specific information regarding the parameter counts of various models in the FB15K-237 and FB15K datasets, please refer to Table 17.

We utilize the 3H-TH model as a reference and set the entity dimensions of 3H-TH to 32. The calculation of entity dimension results, denoted as k^*, for various models in the FB15K-237 and FB15K datasets, along with the link prediction accuracy results of FB15K at different entity dimensions, can be found in Table 18. This ensures that the overall parameters remain the same across the models. The reason for conducting experiments exclusively on FB15K, rather than FB15K-237, is that the calculation entity dimension results for FB15K-237 closely align with 32, as indicated in Table 18. Furthermore, WN18RR exhibits fewer relations (11) and a larger number of entities (40943) compared to FB15K-237. As a result, the calculation entity dimension results for WN18RR are also similar to 32, rendering additional experiments unnecessary. Moreover, we carefully select the appropriate dimensions for each model to ensure the proper functioning of the experiments. For instance, the dimension for 3D rotation must be a multiple of 4, while the dimension for 2D rotation is 2.

Based on the link prediction accuracy results presented in Table 18, it is evident that the 3H model with an entity dimension of $k = 36$ surpasses all other models, including the 3H-TH model. This observation highlights the effectiveness and applicability of the 3H model in KGE tasks.

A.7 State of the art methods in KGE

There are several noteworthy performance methods appeared recently, and we make the following summary for WN18RR in Table 19. Among them, the methods of MoCoSA(He et al., 2023), SimKGC(Wang et al., 2022a), C-LMKE(Wang

Model	Relation embeddings	Num-params	Num-params(FB15K-237)	Num-params(FB15K)
TE	$\mathbf{e}_r$	n_ek+n_rk	$14541k+237k,(14778k)$	$14951k+1345k,(16296k)$
TH	$\mathbf{b}_r$	n_ek+n_rk	$14541k+237k,(14778k)$	$14951k+1345k,(16296k)$
2H or 2E	$\mathbf{c}_r$	$n_ek+n_r\frac{1}{2}k$	$14541k+\frac{237}{2}k,(14660k)$	$14951k+\frac{1345}{2}k,(15624k)$
3H or 3E	$\mathbf{q}_r$	$n_ek+n_r\frac{3}{4}k$	$14541k+\frac{237*3}{4}k,(14719k)$	$14951k+\frac{1345*3}{4}k,(15960k)$
2E-TE	$\mathbf{c}_r,\mathbf{e}_r$	$n_ek+n_r\frac{3}{2}k$	$14541k+\frac{237*3}{2}k,(14897k)$	$14951k+\frac{1345*3}{2}k,(16969k)$
3E-TE	$\mathbf{q}_r,\mathbf{e}_r$	$n_ek+n_r\frac{7}{4}k$	$14541k+\frac{237*7}{4}k,(14956k)$	$14951k+\frac{1345*7}{4}k,(17305k)$
3H-TH	$\mathbf{q}_r,\mathbf{b}_r$	$n_ek+n_r\frac{7}{4}k$	$14541k+\frac{237*7}{4}k,\mathbf{(14956k)}$	$14951k+\frac{1345*7}{4}k,\mathbf{(17305k)}$
2E-TE-2H-TH	$\mathbf{c}_{(r,E)},\mathbf{e}_r,\mathbf{c}_{(r,H)},\mathbf{b}_r$	n_ek+n_r3k	$14541k+237*3k,(15252k)$	$14951k+1345*3k,(18986k)$
3E-TE-3H-TH	$\mathbf{q}_{(r,E)},\mathbf{e}_r,\mathbf{q}_{(r,H)},\mathbf{b}_r$	$n_ek+n_r\frac{7}{2}k$	$14541k+\frac{237*7}{2}k,(15371k)$	$14951k+\frac{1345*7}{2}k,(19659k)$

Table 17: The total number of parameters for several models in the FB15K-237 and FB15K datasets. k denotes entity dimensions, n_e, n_r denotes number of entities and relations.

Model	k^*(FB15K-237)	k^*(FB15K)	experiment-dim (FB15K)	MRR	H@1	H@3	H@10
TransE(TE)	32.4	34	34	.473	.345	.550	.700
RotatE(2E)	32.6	35.4	36	.474	.354	.540	.706
QuatE(3E)	32.5	34.7	36	.494	.370	.569	.721
MuRP(TH)	32.4	34	34	.490	.361	.561	.721
RotH(2H)	32.6	35.4	36	.505	.380	<u>.585</u>	.729
3H	32.5	34.7	36	**.520**	**.395**	**.598**	**.745**
2E-TE	32.1	32.6	32	.494	.373	.568	.725
3E-TE	32	32	32	.496	.376	.572	.725
3H-TH	**32**	**32**	32	<u>.506</u>	<u>.383</u>	.581	<u>.731</u>
2E-TE-2H-TH	31.4	29.2	30	.488	.364	.560	.715
3E-TE-3H-TH	31.1	28.2	28	.477	.355	.548	.704

Table 18: The link prediction accuracy results of FB15K in different entity dimensions. **Bold** indicates the best score, and <u>underline</u> represents the second-best score. k^*(FB15K-237) and k^*(FB15K) are the entity dimensions for several models under the same number of Parameters when we set that of the 3H-TH model as 32, experiment-dim denotes the dimensions that we actually use in experiments for proper experimentation.

et al., 2022b), KNN-KGE(Zhang et al., 2022), and HittER(Chen et al., 2020) are mainly based on Large Language Models to complete the dataset information, thereby achieving better results. LERP(Han et al., 2023) did not use LLMs, but they used some additional contextual information (Logic Rules) beyond the dataset to complete some information missing in the entities and relations. Compared to other methods that rely on the dataset itself, for instance, TransE(Bordes et al., 2013), RotatE(Sun et al., 2019), and the method 3H-TH in this paper, they only used the data and information of the KGE dataset itself, and based on certain mathematical rules and algorithms to get the final result, without using any additional information, and are not similar to LLMs' black box methods. Hence, these dataset-dependent methods continue to hold significant value for KGE research.

A.8 Relation pattern examples

In knowledge graphs (KGs), various relation patterns can be observed, including symmetry, antisymmetry, inversion, composition (both commutative and non-commutative), hierarchy, and multiplicity. These patterns are illustrated in Fig. 3.

Some relations exhibit symmetry, meaning that if a relation holds between entity x and y $((r_1(x,y) \Rightarrow r_1(y,x)))$(e.g., *is married to*), it also holds in the reverse direction (i.e., between y and x). On the other hand, some relations are antisymmetric $((r_1(x,y) \Rightarrow \neg r_1(y,x)))$, where if a relation holds between x and y (e.g., *is father of*), it does not hold in the reverse direction (i.e., between y and x).

Inversion $((r_1(x,y) \Leftrightarrow r_2(y,x)))$ of relations is also possible, where one relation can be transformed into another by reversing the direction of the relation (e.g., *is child of* and *is parent of*).

Composition $((r_1(x,y) \cap r_2(y,z) \Rightarrow r_3(x,z)))$ of relations is another important pattern, where

Model	Description	MRR Accuracy
MoCoSA(He et al., 2023)	Language Models	.696
SimKGC(Wang et al., 2022a)	Language Models	.671
LERP(Han et al., 2023)	Additional Contextual Information (Logic Rules)	.622
C-LMKE(Wang et al., 2022b)	Language Models	.598
KNN-KGE(Zhang et al., 2022)	Language Models	.579
HittER(Chen et al., 2020)	Language Models	.503
3H-TH	-	.493

Table 19: State of the art baseline models in WN18RR dataset.

the combination of two or more relations leads to the inference of a new relation. This composition can be commutative (order-independent) or non-commutative (order-dependent). Non-commutative composition $((r_1(x,y) \cap r_2(y,z) \neq (r_2(x,y) \cap r_1(y,z))$ is necessary when the order of relations matters, such as in the example of the mother of A's father (B) being C and the father of A's mother (D) being E. In a commutative composition, C and E would be equal, but in a non-commutative composition, they are not.

Hierarchical relations exist in KGs, where different entities have different levels or hierarchies. This hierarchical structure is depicted in the tree-like structure shown in Fig. 3.

Finally, multiplicity refers to the existence of different relations between the same entities. For example, an entity can have multiple relations such as *award-winner* and *director* associated with it.

These various relation patterns capture the complexity and diversity of knowledge in KGs, highlighting the challenges and opportunities in modeling and reasoning over such data.

A.9 Hyperparameter

All the hyperparameter settings have been shown in Table 20.

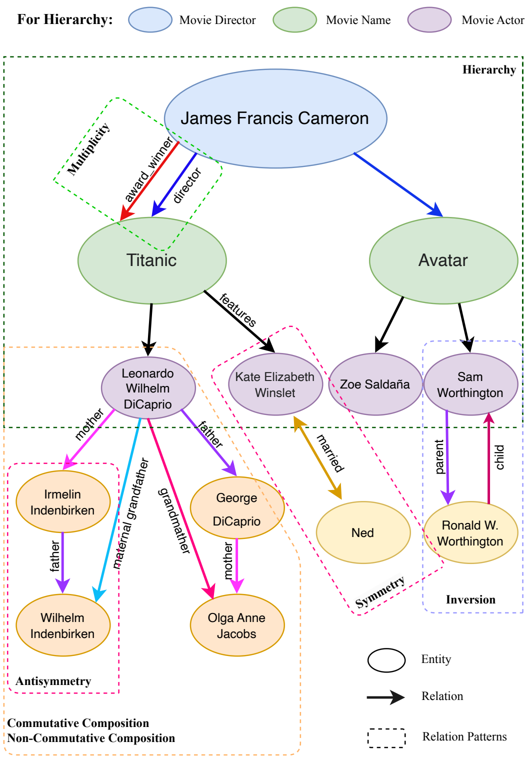

Figure 3: Toy examples for several relation patterns. Our approach can perform well on all these relation patterns.

Dataset	embedding dimension	model	learning rate	optimizer	batch size	negative samples
WN18RR	32	TransE(TE)	0.001	Adam	500	50
		RotatE(2E)	0.1	Adagrad	500	50
		QuatE(3E)	0.2	Adagrad	500	50
		MuRP(TH)	0.0005	Adam	500	100
		RotH(2H)	0.0005	Adam	500	50
		3H	0.001	Adam	500	100
		2E-TE	0.1	Adagrad	500	50
		3E-TE	0.2	Adagrad	500	100
		2E-TE-2H-TH	0.001	Adam	500	100
		3H-TH	0.001	Adam	500	100
		3E-TE-3H-TH	0.001	Adam	500	100
	200	TransE(TE)	0.001	Adam	500	100
		RotatE(2E)	0.1	Adagrad	500	100
		QuatE(3E)	0.2	Adagrad	500	100
		MuRP(TH)	0.001	Adam	500	100
		RotH(2H)	0.001	Adam	500	50
		3H	0.001	Adam	500	100
		2E-TE	0.1	Adagrad	500	50
		3E-TE	0.2	Adagrad	500	100
		2E-TE-2H-TH	0.001	Adam	500	100
		3H-TH	0.001	Adam	500	100
		3E-TE-3H-TH	0.001	Adam	500	100
	300, 500	TransE(TE)	0.001	Adam	500	100
		RotatE(2E)	0.1	Adagrad	500	100
		QuatE(3E)	0.2	Adagrad	500	100
		MuRP(TH)	0.001	Adam	500	100
		RotH(2H)	0.001	Adam	500	50
		3H	0.001	Adam	500	100
		2E-TE	0.1	Adagrad	500	50
		3E-TE	0.2	Adagrad	500	100
		2E-TE-2H-TH	0.001	Adam	500	100
		3H-TH	0.001	Adam	500	100
		3E-TE-3H-TH	0.001	Adam	500	100
FB15k-237	32	TransE(TE)	0.05	Adam	1000	50
		RotatE(2E)	0.05	Adagrad	1000	50
		QuatE(3E)	0.05	Adagrad	1000	50
		MuRP(TH)	0.05	Adagrad	1000	50
		RotH(2H)	0.1	Adagrad	1000	50
		3H	0.05	Adagrad	1000	50
		2E-TE	0.05	Adagrad	1000	50
		3E-TE	0.05	Adagrad	1000	50
		2E-TE-2H-TH	0.05	Adagrad	1000	50
		3H-TH	0.05	Adagrad	1000	50
		3E-TE-3H-TH	0.05	Adagrad	1000	50
FB15K	32	TransE(TE)	0.05	Adagrad	1000	200
		RotatE(2E)	0.4	Adagrad	1000	200
		QuatE(3E)	0.2	Adagrad	1000	200
		MuRP(TH)	0.1	Adagrad	1000	200
		RotH(2H)	0.1	Adagrad	1000	200
		3H	0.2	Adagrad	1000	200
		2E-TE	0.4	Adagrad	1000	200
		3E-TE	0.2	Adagrad	1000	200
		2E-TE-2H-TH	0.2	Adagrad	1000	200
		3H-TH	0.2	Adagrad	1000	200
		3E-TE-3H-TH	0.2	Adagrad	1000	200

Table 20: Best hyperparameters in low- and high-dimensional settings for our approach and several composite models.