Learning Hierarchy-Aware Quaternion Knowledge Graph Embeddings with Representing Relations as 3D Rotations

Jinfa Yang, Xianghua Ying*, Yongjie Shi,
Xin Tong, Ruibin Wang, Taiyan Chen, Bowei Xing
Key Laboratory of Machine Perception (MOE)
School of Intelligence Science and Technology, Peking University
{jinfayang, xhying, shiyongjie, xin_tong, robin_wang}@pku.edu.cn,
chenty@stu.pku.edu.cn, 2017xbw@pku.edu.cn

Abstract

Knowledge graph embedding aims to represent entities and relations as low-dimensional vectors, which is an effective way for predicting missing links. It is crucial for knowledge graph embedding models to model and infer various relation patterns, such as symmetry/antisymmetry. However, many existing approaches fail to model semantic hierarchies, which are common in the real world. We propose a new model called HRQE, which represents entities as pure quaternions. The relational embedding consists of two parts: (a) Using unit quaternions to represent the rotation part in 3D space, where the head entities are rotated by the corresponding relations through Hamilton product. (b) Using scale parameters to constrain the modulus of entities to make them have hierarchical distributions. To the best of our knowledge, HRQE is the first model that can encode symmetry/antisymmetry, inversion, composition, multiple relation patterns and learn semantic hierarchies simultaneously. Experimental results demonstrate the effectiveness of HRQE against some of the SOTA methods on four well-established knowledge graph completion benchmarks.

1 Introduction

Knowledge graphs represent human knowledge of the real world as structured triples— (head entity, relation, tail entity) also known as (subject, predicate, object). There are some outstanding knowledge graphs, such as WordNet (Miller, 1995), Freebase (Bollacker et al., 2008), DBpedia (Lehmann et al., 2015). They have gained widespread attention for their successful usage in various applications (*e.g.*, question answering, natural language processing, and recommendation systems). Although millions of entities and billions of facts exist in large-scale knowledge graphs, they still suffer from the incompleteness problem. Therefore, knowledge graph completion (also known as link prediction) which aims to predict missing links among entities based on the known triples has gained growing interest. Learning low-dimensional representations of entities and relations for Knowledge graphs is an effective solution for this task.

Learning knowledge graph embeddings in the complex space $\mathbb{C}$ or quaternion space $\mathbb{H}$ has been proven to be a highly effective inductive bias, largely owing to their ability to model connectivity patterns of the relations. For example, ComplEx (Trouillon et al., 2016), which infers new relational triplets with the asymmetrical Hermitian product can model the symmetry/antisymmetry patterns. RotatE (Sun et al., 2019), which represents entities as points in a complex space and relations as rotations, can model relation patterns including symmetry/antisymmetry, inversion, and composition. DualE (Cao et al., 2021), which combines rotation and translation in dual quaternion space can additionally model the multiple relations pattern. However, many existing models fail to model semantic hierarchies in knowledge graphs.

Semantic hierarchy is a ubiquitous property in knowledge graphs. For instance, WordNet contains the triple [arbor/cassia/palm, hypernym, tree], where "tree" is at a higher level than "arbor/cassia/palm" in the hierarchy. Freebase contains the triple [America, /location/location/contains, California/Los Angeles], where "California/Los Angeles" is at a lower level than "America" in the hierarchy. Although there exists some work that takes the hierarchy structures into account (Xie et al., 2016; Zhang et al., 2020), they usually require additional data to obtain the hierarchy information or cannot model various relation patterns. Therefore, it is still challenging to find an approach that is capable of modeling the various relation patterns and semantic hierarchy simultaneously.

[*] Corresponding Author

Proceedings of the 29th International Conference on Computational Linguistics, pages 2011–2023
October 12–17, 2022.

Model	Relation Patterns					Hierarchy
	Multiple	Symmetry	Antisymmetry	Inversion	Composition	-Aware
TransE (Bordes et al., 2013)	✗	✗	✓	✓	✓	✗
DistMult (Yang et al., 2015)	✗	✓	✗	✗	✗	✗
ComplEx (Trouillon et al., 2016)	✗	✓	✓	✓	✗	✗
RotatE (Sun et al., 2019)	✗	✓	✓	✓	✓	✗
QuatE (Zhang et al., 2019)	✗	✓	✓	✓	✗	✗
HAKE (Zhang et al., 2020)	✗	✓	✓	✓	✓	✓
DualE (Cao et al., 2021)	✓	✓	✓	✓	✓	✗
QuatRE (Nguyen et al., 2022)	✓	✓	✓	✓	✗	✗
RQE	✓	✓	✓	✓	✓	✗
HRQE	✓	✓	✓	✓	✓	✓

Table 1: The pattern modeling and hierarchy-aware abilities of several models

In this paper, we propose **R**otation Based **Q**uaternion Knowledge Graph **E**mbeddings (RQE) and its **H**ierarchy-aware extension HRQE. More concretely, we represent entities as pure quaternions with three imaginary components **i**, **j** and **k**. The relational embedding consists of two parts: (a) Using unit quaternions to represent the rotation part in 3D space, where the head entities Q_h are rotated by the corresponding relations through Hamilton product. (b) Using scale parameters to constrain the modulus of entities Q_h and Q_t to make them have hierarchical distributions.

To summarize, our contributions are as follows: 1) We propose a new framework called HRQE based on quaternion rotation. 2) To the best of our knowledge, HRQE is the first model that can encode symmetry/antisymmetry, inversion, composition, multiple relation patterns and learn semantic hierarchies simultaneously. 3) We conduct a series of theoretical and empirical analyses to show the strength of HRQE against some of the SOTA methods.

2 Related Work

2.1 Knowledge Graph Embedding Models

Roughly speaking, we can divide knowledge graph embedding models into translational distance models and semantic matching models. The former use distance-based score functions, while the latter use similarity-based ones.

Translational Distance Models. TransE (Bordes et al., 2013) is the most widely used translation distance constraint model. It assumes that entities and relations satisfy $head + relation \approx tail$. However, TransE cannot handle 1-1-N, N-1-1, and N-1-N relations well (Wang et al., 2014). TransH (Wang et al., 2014) is proposed to compensate for the shortcomings of TransE. It projects entities onto relation-specific hyperplanes. TransR (Lin et al., 2015) has a very similar idea to TransH, which introduces relation-specific spatial transformations instead of hyperplanes. TranSparse (Ji et al., 2016) simplifies TransR by forcing the projection matrix to be sparse. Moreover, RotatE (Sun et al., 2019) defines each relation as a rotation from the source entity to the target entity in a complex vector space, which can represent various relation patterns including symmetry/asymmetry, inversion and composition.

Semantic Matching Models. RESCAL (Nickel et al., 2011) is a tensor factorization model which represents each relation as a full-rank matrix and obtains score function by matrix multiplication. DistMult (Yang et al., 2015) simplifies RESCAL by restricting relation matrices to be diagonal. However, Distmult assumes that all relations are symmetric. ComplEx (Trouillon et al., 2016) extends DistMult to complex space, and uses conjugate-transpose to model asymmetric relations. QuatE (Zhang et al., 2019) extends the complex space into the quaternion space with two rotating surfaces. DualE (Cao et al., 2021) combines rotation and translation in dual quaternion space. ConvE (Dettmers et al., 2018) and InteractE (Vashishth et al., 2020) employ convolutional neural networks to build score functions.

2.2 The Ways to Model Hierarchy Structures

Another related problem is how to model hierarchy structures in knowledge graphs. Xie et al. (2016) propose TKRL, which requires additional hierarchical type information for entities. Zhang et al. (2018) use clustering algorithms to model the hierarchical relation structures. Zhang et al.

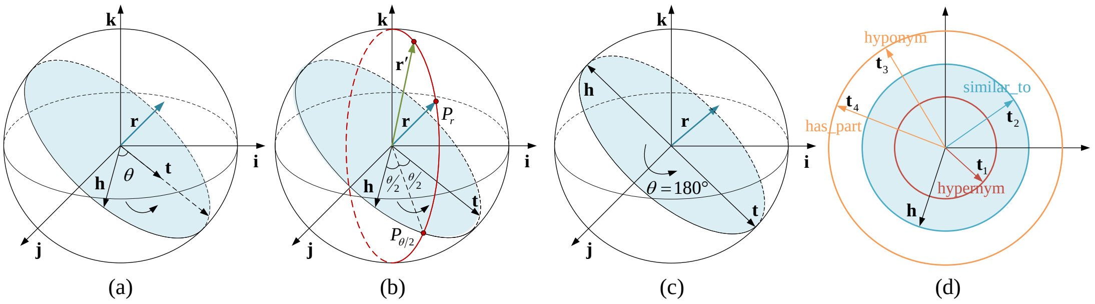


Figure 1: Illustrations of HRQE. (a) HRQE models r as rotation in 3D space. (b) Example of HRQE modeling multiple relations. (c) An example HRQE of modeling symmetric relations with $\theta = 180°$. (d) Example of HRQE modeling different levels of the hierarchy.

(2020) proposed HAKE, which maps entities into the polar coordinate system for hierarchy-aware. Inspired by HAKE, we project entities into 3D space and constrain their rotations and modulus with corresponding relations. In addition to learning the semantic hierarchy, we can better encode various relation patterns such as multiple relations.

3 Quaternion Background

A quaternion $Q \in \mathbb{H}$ is a hyper-complex number consisting of a real and three separate imaginary components (Hamilton, 1844), defined as $Q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$, where $a, b, c, d \in \mathbb{R}$ and $\mathbf{i}, \mathbf{j}, \mathbf{k}$ are imaginary units. $\mathbf{i}, \mathbf{j}$ and $\mathbf{k}$ are satisfied with Hamilton's rules ($\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1$). And based on these rules, more non-commutative multiplication rules can be derived, such as $\mathbf{ij} = \mathbf{k}, \mathbf{ji} = -\mathbf{k}, \mathbf{jk} = \mathbf{i}, \mathbf{kj} = -\mathbf{i}, \mathbf{ki} = \mathbf{j}$, and $\mathbf{ik} = -\mathbf{j}$. Some widely used operations of quaternion algebra are introduced as follows:

Quaternion Conjugate: The conjugate of a quaternion Q is defined as $\bar{Q} = a - b\mathbf{i} - c\mathbf{j} - d\mathbf{k}$.

Quaternion Norm: The norm of a quaternion Q is defined as $|Q| = \sqrt{a^2 + b^2 + c^2 + d^2}$.

Pure Quaternion: A pure quaternion $Q \in \mathbb{H}_p$ is defined as a quaternion whose scalar part is zero. Usually, we convert the 3D space point (x, y, z) into a pure quaternion $Q = 0 + x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$, x, y and $z \in \mathbb{R}$ for further quaternion operations.

Quaternion-Inner Product: The quaternion inner product between $Q_1 = a_1 + b_1\mathbf{i} + c_1\mathbf{j} + d_1\mathbf{k}$ and $Q_2 = a_2 + b_2\mathbf{i} + c_2\mathbf{j} + d_2\mathbf{k}$ is obtained by taking the inner products between corresponding scalar and imaginary components and returns a scalar $Q_1 \cdot Q_2 = \langle a_1, a_2 \rangle + \langle b_1, b_2 \rangle + \langle c_1, c_2 \rangle + \langle d_1, d_2 \rangle$.

Quaternion Multiplication (Hamilton Product): The quaternion multiplication is composed of all the standard multiplications of factors in quaternions and returns another quaternion, defined as:

$$\begin{aligned} Q_1 \otimes Q_2 = {} & (a_1 a_2 - b_1 b_2 - c_1 c_2 - d_1 d_2) \\ & + (a_1 b_2 + b_1 a_2 + c_1 d_2 - d_1 c_2)\mathbf{i} \\ & + (a_1 c_2 - b_1 d_2 + c_1 a_2 + d_1 b_2)\mathbf{j} \\ & + (a_1 d_2 + b_1 c_2 - c_1 b_2 + d_1 a_2)\mathbf{k}. \end{aligned} \tag{1}$$

Quaternion Rotation: If $Q_r = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}\mathbf{u}$, where $\mathbf{u} \in \mathbb{R}\mathbf{i} + \mathbb{R}\mathbf{j} + \mathbb{R}\mathbf{k}$ is a unit vector, the result of pure quaternion $Q = 0 + x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$ rotating θ around the axis $\mathbf{u}$ is $Q' = 0 + x'\mathbf{i} + y'\mathbf{j} + z'\mathbf{k}$, then

$$Q' = Q_r \otimes Q \otimes \bar{Q}_r. \tag{2}$$

4 Method

In this section, we introduce our proposed model HRQE. First of all, we elaborate the details of our framework, which mainly consists of two parts: (1) rotate the head entity using the unit relation quaternion and score each triplet with inner product between the rotated head quaternion and the tail quaternion; (2) limit the norm of the head quaternion and the tail quaternion with the relation modulus part. After that, we provide a series of analyses to show the strength of our framework.

Symbol Description. Suppose that we have a knowledge graph $\mathcal{G}$ consisting of N entities and M relations. We formulate the all entity embeddings as a pure quaternion matrix $Q \in \mathbb{H}_p^{N \times k}$, where each row is an embedding vector for a specific entity of dimensionality k, and denote the relation embeddings as rotation part $W \in \mathbb{H}^{M \times k}$ and modulus part $w \in \mathbb{R}^{M \times k}$. Given a triplet (h, r, t), the embedding of head entity h is denoted as $Q_h = \{0 + x_h\mathbf{i} + y_h\mathbf{j} + z_h\mathbf{z} : x_h, y_h, z_h \in \mathbb{R}^k\}$ and the embedding of the tail entity $Q_t = \{0 + x_t\mathbf{i} + y_t\mathbf{j} + z_t\mathbf{z} : x_t, y_t, z_t \in \mathbb{R}^k\}$, where $Q_h, Q_t \in Q$.

Then we denote the relation r as rotation part $W_r = \{a_r + b_r\mathbf{i} + c_r\mathbf{j} + d_r\mathbf{z} : a_r, b_r, c_r, d_r \in \mathbb{R}^k\}$ and modulus part $w_r = \{e_r : e_r \in \mathbb{R}^k\}$, where $W_r \in W, w_r \in w$.

4.1 Hierarchy-Aware Rotation Quaternion Embeddings

The Rotation Part. We first normalize the relation quaternion W_r to a unit quaternion $W_r^{\triangleleft}$ to eliminate the scaling effect by dividing W_r by its norm:

$$W_r^{\triangleleft} = \frac{W_r}{|W_r|} = \frac{a_r + b_r\mathbf{i} + c_r\mathbf{j} + d_r\mathbf{k}}{a_r^2 + b_r^2 + c_r^2 + d_r^2}. \tag{3}$$

Secondly, we rotate the head entity Q_h by doing Hamilton product with $W_r^{\triangleleft}$ and $\bar{W}_r^{\triangleleft}$:

$$Q'_h(r'_h, x'_h, y'_h, z'_h) = W_r^{\triangleleft} \otimes Q_h \otimes \bar{W}_r^{\triangleleft}, \tag{4}$$

where $\otimes$ denotes the element-wise multiplication between two vectors. Then the rotation part scoring function $\phi_r(h, r, t)$ is defined by the quaternion inner product:

$$\phi_r(h, r, t) = Q'_h \cdot Q_t = \langle x'_h, x_t \rangle + \langle y'_h, y_t \rangle + \langle z'_h, z_t \rangle. \tag{5}$$

We separate the rotation part as an independent model RQE, which achieves impressive results (refer to Section 5).

The Modulus Part. As shown in Figure 1a, the rotation part allows the head entity to rotate in 3D space to approximate the tail entity. The modulo length of entities is used to represent the hierarchical distribution of entities. The modulus part of relations is used to measure the hierarchical difference between head and tail entities, which is beneficial for learning hierarchy-aware, see Section 4.2 for details. The modulus part scoring function $\phi_m(h, r, t)$ is defined as:

$$\begin{aligned}\phi_m(h,r,t) &= -\| w_r |Q_h| - |Q_t| \|_1 \\ &= -\left\| w_r \sqrt{x_h^2 + y_h^2 + z_h^2} - \sqrt{x_t^2 + y_t^2 + z_t^2} \right\|_1.\end{aligned} \tag{6}$$

Finally, The scoring function of HRQE is:

$$\phi(h, r, t) = \phi_r(h, r, t) + \lambda \phi_m(h, r, t), \tag{7}$$

where $\lambda \in \mathbb{R}$ is a parameter that learned by the model.

Loss Function. Following Trouillon et al. (2016), We formulate the task as a classification problem and adopt the cross-entropy loss as our loss function. Ω and $\Omega' = \mathcal{E} \times \mathcal{R} \times \mathcal{E} - \Omega$ are used to denote the set of observed triplets and the set of unobserved triplets, respectively. Moreover, we use the ℓ_2 norm with regularization rates λ_1 and λ_2 to regularize Q and W:

$$\begin{aligned} L = \sum_{r(h,t) \in \Omega \cup \Omega^-} & \log(1 + \exp(-Y_{hrt}\phi(h, r, t))) \\ & + \lambda_1 \|Q\|_2^2 + \lambda_2 \|W\|_2^2, \end{aligned} \tag{8}$$

where $\Omega^- \subset \Omega'$ with negative sampling strategies such as uniform sampling, bernoulli sampling (Wang et al., 2014), and adversarial sampling (Sun et al., 2019). $Y_{hrt} \in \{-1, 1\}$ represents the corresponding label of the triplet (h, r, t). We optimize the loss function by utilizing Adagrad (Duchi et al., 2011).

4.2 Discussion

In this part, we discuss the theoretical properties of HRQE. We summarize several popular knowledge graph embedding models in Appendix A.5 and definitions of various relation patterns in Appendix A.1.

Capability in Modeling Multiple Relations. For multiple relations such as (A, classmate, B) and (A, neighbor, B) $\in \mathcal{G}$, HRQE allows multiple expressions for the relations when the head and tail entities are fixed. As shown in Figure 1b (here the modulus part is simplified as $w_r = 1$), the red arc passes through $\mathbf{r}$ vertex P_r and the angle bisector vertex $P_{\theta/2}$. $\mathbf{r'}$ with vertex on the red arc can also make $\mathbf{h}$ rotate to $\mathbf{t}$. That is, $\exists r' \neq r$, (h, r, t) and $(h, r', t) \in \mathcal{G}$.

Capability in Modeling Symmetry /Antisymmetry, Inversion and Composition. The flexibility and representational power of quaternion rotation enable us to model various relation patterns at ease. Since HRQE degenerates to RQE when $\lambda = 0$, we mainly use RQE to discuss. When the rotation angle $\theta = [0°, 180°, 270°, 360°]$, RQE can model the symmetry pattern, and when $W_{r1}^{\triangleleft} = \bar{W}_{r2}^{\triangleleft}$, RQE can model the inversion pattern. The specific lemmas and proofs are as follows:

Lemma 1 *HRQE can infer the symmetry /antisymmetry pattern. (See proof in Appendix A.2)*

Models	WN18					FB15K				
			Hits(%)					Hits(%)		
	MR	MRR(%)	@10	@3	@1	MR	MRR	@10	@3	@1
TransE	-	49.5	94.3	88.8	11.3	-	46.3	74.9	57.8	28.7
DistMult	655	79.7	94.6	-	-	42.2	79.8	89.3	-	-
HolE	-	93.8	94.9	94.5	93.0	-	52.4	73.9	75.9	59.9
ComplEx	-	94.1	94.7	94.5	93.6	-	69.2	84.0	75.9	59.9
ConvE	374	94.3	95.6	94.6	93.5	51	65.7	83.1	72.3	55.8
R-GCN+	-	81.9	96.4	92.9	69.7	-	69.6	84.2	76.0	60.1
SimplE	-	94.2	94.7	94.4	93.9	-	72.7	83.8	77.3	66.0
NKGE	366	94.7	95.7	94.9	94.2	56	73	87.1	79.0	65.0
TorusE	-	94.7	95.4	95.0	94.3	-	73.3	83.2	77.1	67.4
RotatE	184	94.7	96.1	95.3	93.8	32	69.9	87.2	78.8	58.5
a-RotatE	309	94.9	95.9	95.2	94.4	40	79.7	88.4	83.0	74.6
QuatE	162	95.0	95.9	95.4	**94.5**	**17**	78.2	**90.0**	83.5	71.1
Rotat3D	214	95.1	96.1	95.3	**94.5**	39	78.9	88.7	83.5	72.8
RQE	117	**95.2**	96.3	**95.6**	**94.5**	23	**81.3**	89.2	**84.3**	**76.6**
HRQE	**72**	**95.2**	**96.5**	**95.6**	94.4	32	79.1	88.7	83.7	73.0

Table 2: Evaluation results on WN18, FB15k datasets. The best scores are in bold, while the second best scores are in underline.

Lemma 2 *HRQE can infer the inversion pattern. (See proof in Appendix A.3)*

Lemma 3 *HRQE can infer the composition pattern. (See proof in Appendix A.4)*

Capability in Modeling Hierarchy Structures. To model the semantic hierarchies of knowledge graphs, a knowledge graph embedding model must be capable of distinguishing entities in the following two categories. (a) Entities at the same level of the hierarchy. (*e.g.* "truck" and "lorry") (b) Entities at different levels of the hierarchy. (*e.g.* "mammal" and "cat") (Zhang et al., 2020). For HRQE, the rotate part can model the entities at the same level of the semantic hierarchy, and the modulus part can model the entities at different levels of the hierarchy. As shown in Figure 1d, entities have hierarchical distribution under different relations, and we simplify it into a 2D space for display. That is, HRQE maps entities into the 3D polar coordinate system, where the angular coordinates and the radial coordinates correspond to the rotate part and the modulus part, respectively.

These results are also summarized in Table 1. We can see that HRQE is the only model that can model and infer all types of relation patterns and hierarchy awareness.

5 Experiments and Results

5.1 Experimental Setup

We evaluate our proposed models on four widely used benchmarks, which are statistically summarized in Table 4.

Datasets: WN18 (Bordes et al., 2013) is extracted from WordNet (Miller, 1995), a database featuring lexical relations and conceptual-semantic between words. The dataset has many inverse relations and the mainly relation patterns are **symmetric/antisymmetric** and **inversion**. WN18RR (Dettmers et al., 2018) is a subset of WN18, with inverse relations removed. The main relation patterns are **symmetric/antisymmetric** and **composition**. In WN18 and WN18RR, most of the triples consist of hyponym and hypernym relations which make them tend to follow a strictly **hierarchical structure**. FB15K (Bordes et al., 2013) is extracted from Freebase (Bollacker et al., 2008), which is a large-scale knowledge graph containing general human knowledge. The key of link prediction on FB15k is to model and infer the **symmetry/antisymmetry** and **inversion** patterns. FB15K-237 (Toutanova and Chen, 2015) is a subset of FB15K, with inverse relations removed. Therefore, the key to link prediction on FB15K-237 boils down to model and infer **symmetrical/antisymmetric** and **composition** patterns.

Evaluation Protocol: For each triple (h, r, t) in the test dataset, we replace either the head entity h or the tail entity t with the total list of the embedding entities. Then, we base the score function to rank the candidate entities in descending order. The filtered setting is used to remove some correct results that appear in the training set or valida-

Models	WN18RR					FB15K-237				
			Hits(%)					Hits(%)		
	MR	MRR(%)	@10	@3	@1	MR	MRR	@10	@3	@1
TransE	3384	22.6	50.1	-	-	357	29.4	46.5	-	-
DistMult	5100	43	49	44	39	254	24.1	41.9	26.3	15.5
ComplEx	5261	44	51	46	41	339	24.7	42.8	27.5	15.8
ConvE	4187	43	52	44	40	244	32.5	50.1	35.6	23.7
InteractE	5202	46.3	52.8	-	43.0	172	35.4	53.5	-	26.3
RotatE	3277	47.0	56.5	48.8	42.2	185	29.7	48.0	32.8	20.5
a-RotatE	3340	47.6	57.1	49.2	42.8	177	33.8	53.3	37.5	24.1
QuatE	2314	48.8	58.2	50.8	43.8	87	34.8	55.0	38.2	24.8
ComplEx-N3	-	48.0	57.2	49.5	43.5	-	35.7	54.7	39.2	26.4
TuckER	-	47.0	52.6	48.2	44.3	-	35.8	54.4	39.4	26.6
MURP	-	47.5	55.4	48.7	43.6	-	33.6	52.1	37.0	24.5
RoTH	2293	49.1	58.6	51.1	44.1	-	34.4	53.5	38.0	24.6
Rotat3D	3328	48.9	57.9	50.5	44.2	165	34.7	54.3	38.5	25.0
HAKE	-	49.7	58.2	51.6	45.2	-	34.6	54.2	38.1	25.0
DualE	2270	49.2	58.4	51.3	44.4	91	36.5	55.9	40.0	26.8
QuatRE	1986	49.3	59.2	51.9	43.9	88	36.7	56.3	40.4	26.9
RQE	2043	49.7	59.1	51.7	44.8	**86**	36.9	56.8	40.4	27.3
HRQE	**1198**	**50.5**	**60.1**	**52.4**	**45.4**	89	**37.2**	**56.9**	**40.7**	**27.5**

Table 3: Evaluation results on WN18RR, FB15k-237 datasets. The best scores are in bold, while the second best scores are in underline.

Dataset	#En	#Re	#train	#valid	#test
WN18	40,943	18	141,442	5,000	5,000
WN18RR	40,943	11	86,835	3,034	3,134
FB15K	14,951	1,345	483,142	50,000	59,071
FB15k-237	14,541	237	272,115	17,535	20,466

Table 4: Number of entities, relations, and observed triplets in each split for benchmarks.

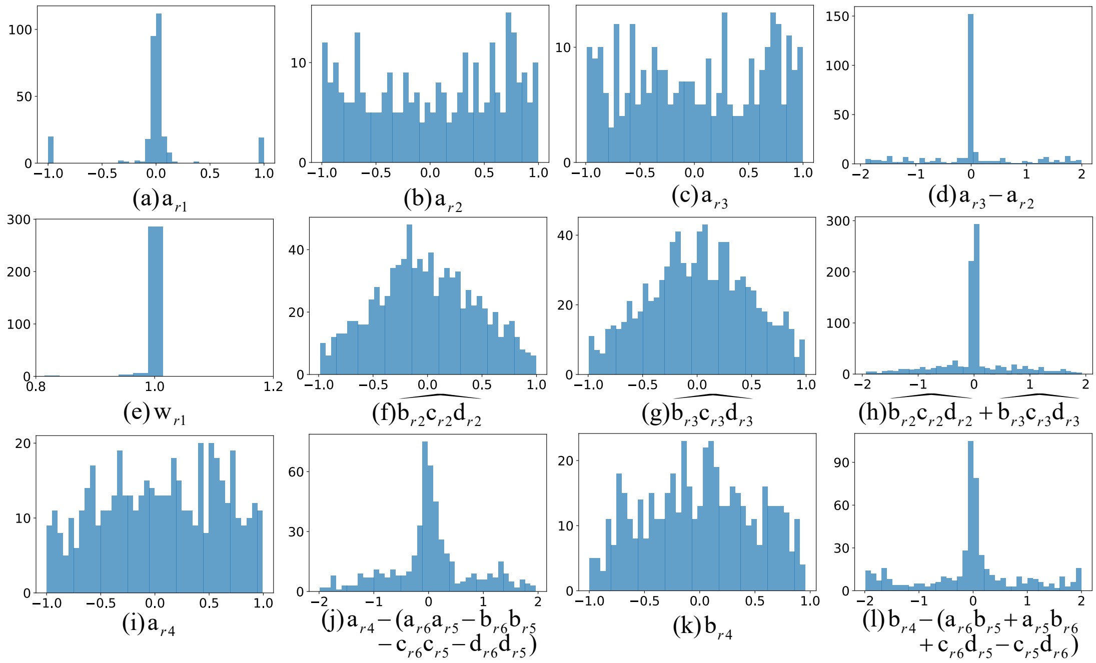
tion set but not in test set. We choose Mean Rank (MR), Mean Reciprocal Rank (MRR) and Hits at N (H@N) as the evaluation metrics. MR measures the average rank of all correct entities with a lower value representing better performance. MRR is the average inverse rank for correct entities. Hit@n measures the proportion of correct entities in the top n entities.

Baselines: We compare HRQE with a number of strong baselines. For Translational Distance Models, we reporte TransE (Bordes et al., 2013) , TorusE (Ebisu and Ichise, 2018), RotatE (Sun et al., 2019), Rotat3D (Gao et al., 2020), ROTH (Chami et al., 2020) and HAKE (Zhang et al., 2020); For Semantic Matching Models, we reporte DistMult (Yang et al., 2015), HolE (Nickel et al., 2016), ComplEx (Trouillon et al., 2016), ComplEx-N3 (Lacroix et al., 2018), SimplE (Kazemi and Poole, 2018), TuckER (Balažević et al., 2019), ConvE (Dettmers et al., 2018), R-GCN (Schlichtkrull et al., 2018), NKGE (ConvE based) (Wang et al., 2018), InteractE (Vashishth et al., 2020), QuatE (Zhang et al., 2019), DualE (Cao et al., 2021), and QuatRE (Nguyen et al., 2022).

Implementation Details: The best models are selected by early stopping on the validation set with Hits@10. The ranges of the hyperparameters for the grid search are set as follows: The embedding size k is selected in $\{100, 200, 300, 400, 500\}$. The regularization rates λ_1 and λ_2 are adjusted in $\{0, 0.01, 0.02, 0.03, 0.05, 0.1, 0.2, 0.3, 0.5\}$. The learning rate is chosen from $\{0.01, 0.02, 0.05, 0.1\}$, the number of negative triples sampled per training triple is selected from $\{1, 2, 5, 10\}$. In addition, we create $\{10, 100\}$ batches of training samples for the different datasets. We report RQE and HRQE with type constraints (Krompaß et al., 2015). The training strategies of self-adversarial negative sampling (Sun et al., 2019) and N3 regularization with reciprocal learning (Lacroix et al., 2018) are not used for RQE and HRQE. All hyper-parameters of our models are provided in the appendix A.6, and our code is available at https://github.com/Jinfa/HRQE.

5.2 Results

The empirical results on four datasets are reported in Table 2 and Table 3. HRQE performs extremely competitively compared to the existing state-of-the-art models across all metrics. As a rotation-



(a) a_{r1} (b) a_{r2} (c) a_{r3} (d) $a_{r3} - a_{r2}$

(e) w_{r1} (f) $\widehat{b_{r2}c_{r2}d_{r2}}$ (g) $\widehat{b_{r3}c_{r3}d_{r3}}$ (h) $\widehat{b_{r2}c_{r2}d_{r2}} + \widehat{b_{r3}c_{r3}d_{r3}}$

(i) a_{r4} (j) $a_{r4} - (a_{r6}a_{r5} - b_{r6}b_{r5} - c_{r6}c_{r5} - d_{r6}d_{r5})$ (k) b_{r4} (l) $b_{r4} - (a_{r6}b_{r5} + a_{r5}b_{r6} + c_{r6}d_{r5} - c_{r5}d_{r6})$

Figure 2: Histograms of relation embeddings for different relation patterns. The corresponding relation is as follows: r_1 is *similar_to*; r_2 is *has_part*; r_3 is *part_of*; r_4 is */location/administrative_division/capital /location/administrative_division_capital_relationship/capital*; r_5 is */location/hud_county_place/place*; r_6 is *base/areas/schema/administrative_area/capital*.

based model, HRQE outperforms the two representative rotation models RoatE, and Roata3D. As a hierarchy-sensitive model, HRQE outperforms the representative hierarchy-aware model HAKE. Also, we outperform other quaternion-valued models such as QuatE, DualE, and QuatRE.

On the WN18 dataset, HRQE outperforms all the baselines on all metrics except Hit@1. HRQE achieves slightly lower results on H@1 than QuatE and RotatE, but surpasses them on the other four metrics, especially on MR with a 56% improvement over QuatE. RQE outperforms HRQE on the FB15K dataset, while the results of them on the validation set are close with 87%. We speculate that excessive inverse relations in FB15K affect the expression of HRQE hierarchy-aware modules. The other recent models a-RotatE, QuatE, and Rotat3D achieve comparable results.

As shown in Table 3, HRQE achieves the best performance over existing state-of-the-art models on the two datasets where trivial inverse relations are removed. On WN18RR in which there are a number of symmetry relations, TransE cannot learn the symmetric relation pattern, so it performs not well. In contrast, the rotation family can achieve good results, and HRQE has further refreshed the performance to achieve the optimal. In addition, HRQE's performance on MR is also impressive, with a 48% improvement over QuatE. WN18 and WN18RR contain hyponym and hypernym relations which make them tend to follow a strictly hierarchical structure, and HRQE's performance demonstrates its ability to learn hierarchically. On FB15K-237, HRQE also achieved better results compared with the previous state-of-the-art models, which shows that HRQE can learn the composition relation pattern better.

Models	Prediction Head (MRR)		Prediction Tail (MRR)	
	1-1-1	1-N-1	1-1-1	1-N-1
TransE	44.3	48.4	45.6	46.9
RotatE	66.8	79.4	71.6	78.6
RQE	**67.8**	**87.7**	**72.5**	**89.5**

Table 5: Evaluation results of multiple relations on FB15k dataset.

5.3 Model Analysis

Analysis on Multiple Relations. In the test set of FB15K, 38121 are single-relation triples (1-1-1), and 20950 are multi-relation triples (1-N-1). To avoid the influence of the modulus part, we choose RQE as the comparison model. Table 5 shows that RQE can better deal with multi-relational triples than TransE and RotatE.

Visualize Some Typical Relation Patterns. To

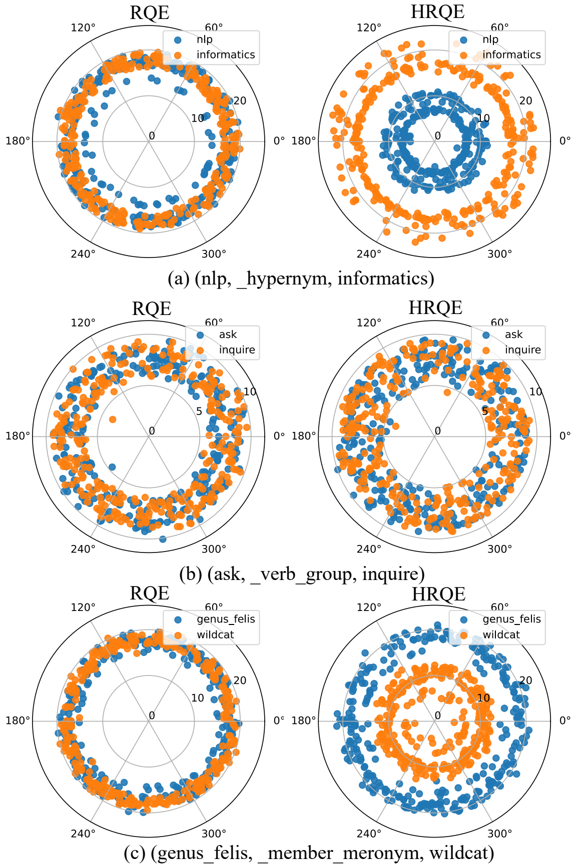
Models	Prediction Head (Hits@10)				Prediction Tail (Hits@10)			
	1-1-1	1-1-N	N-1-1	N-1-N	1-1-1	1-1-N	N-1-1	N-1-N
QuatE	54.2	**66.4**	38.6	46.9	53.1	25.5	88.3	60.9
QuatRE	58.9	**66.4**	39.3	48.1	59.9	26.8	**88.9**	61.7
HRQE	**63.5**	**66.4**	**42.5**	**48.7**	**63.0**	**28.1**	**88.9**	**62.2**

Table 6: Evaluation results of complex relations on FB15k-237 dataset. The first two rows are taken from (Nguyen et al., 2022)

further verify the learned relation patterns, we visualize some examples. For **symmetry pattern**, HRQE encode with rotation angle $\theta = [0°, 180°, 360°]$ (correspondingly $a_{r1} = \cos\frac{\theta}{2} = [-1, 0, 1]$) and modulo weight $w_r = 1$ which are shown in Figure 2 a and e. For **inversion pattern**, we have $W^{\triangleleft}_{r2} = \bar{W}^{\triangleleft}_{r3}$ (correspondingly $a_{r2} - a_{r3} = 0$ and $\widehat{b_{r2}c_{r2}d_{r2}} + \widehat{b_{r3}c_{r3}d_{r3}} = 0$) , which are shown in Figure 2 b,c,d and f,g,h. For **composition pattern**, we have $W^{\triangleleft}_{r4} = W^{\triangleleft}_{r6} \otimes W^{\triangleleft}_{r5}$. We show the real part and the first imaginary part(correspondingly $a_{r4} - (a_{r6}a_{r5} - b_{r6}b_{r5} - c_{r6}c_{r5} - d_{r6}d_{r5}) = 0$ and $b_{r4} - (a_{r6}b_{r5} + a_{r5}b_{r6} + c_{r6}d_{r5} - c_{r5}d_{r6}) = 0$) in Figure 2 i, j, k and l. Table 7 summarizes the MRR for each relation on WN18RR, confirming the superior representation capability of HRQE in modelling different types of relation.

Analysis on Hierarchy-Aware. We plot the entity embeddings of two models: RQE and HRQE. Their entities are all pure quaternions. For an intuitive display, we project it to a 2D plane and display them in polar coordinates. The radius r of the polar coordinates is quaternion norm $|Q|$, and the angle is twice the angle between the entities and $\mathbf{i} + \mathbf{j} + \mathbf{k}$. Note that we use the logarithmic scale to better display the differences between entity embeddings. As all the moduli have values less than one, after applying the logarithm operation, the larger radii in the figures will actually represent smaller modulus. Compared with the tail entities, the head entities in Figure 3 a, b, and c are at lower levels, similar levels, and higher levels in the semantic hierarchy, respectively. We can see that there exist clear hierarchies in HRQE, which demonstrates that the modulus part in HRQE can help effectively model the semantic hierarchies.

Analysis on Complex Relations. We also conduct further investigation on the performance of HRQE on complex relations: 1-1-N, N-1-1, and N-1-N relations. We compare with QuatE and QuatRE (Nguyen et al., 2022). QuatRE adds two additional relational quaternions and quaternion mul-



(a) (nlp, _hypernym, informatics)

(b) (ask, _verb_group, inquire)

(c) (genus_felis, _member_meronym, wildcat)

Figure 3: Visualization of the embeddings of several entity pairs from WN18RR dataset.

Relation Name	RotatE	QuatE	HRQE
hypernym	14.8	17.3	**19.3**
derivationally_related_form	94.7	95.3	**95.7**
instance_hypernym	31.8	36.4	**38.1**
also_see	58.5	62.9	**68.3**
member_meronym	23.2	23.2	**27.2**
synset_domain_topic_of	34.1	46.8	**48.4**
has_part	18.4	23.3	**24.3**
member_of_domain_usage	31.8	**44.1**	42.6
member_of_domain_region	20.0	19.3	**27.1**
verb_group	**94.3**	92.4	91.1
similar_to	**100.0**	**100.0**	**100.0**

Table 7: MRR for the models tested on each relation of WN18RR.

tiplies with the head and tail entities to improve QuatE's ability for handling complex relations. Table 6 shows the MRR and H@10 scores for predicting the head entities and then the tail entities with respect to each relation category on FB15k-237, wherein our HRQE outperforms QuatE and QuatRE on these relation categories.

6 Conclusion

To model various relation patterns and semantic hierarchies in knowledge graphs, we propose a novel knowledge graph embedding model HRQE, which maps entities into 3D space with rotation and modulo constraints. Empirical experimental evaluations on benchmark datasets show that our proposed HRQE significantly outperforms several existing state-of-the-art methods. Further investigation shows that HRQE is capable of modeling relations with various relation patterns and modeling entities at both different levels and the same levels in the semantic hierarchies.

Acknowledgement

This work was supported in part by National Key R&D Program of China Grand No. 2020YFB1708002, and NNSFC Grant No. 61971008.

References

Ivana Balažević, Carl Allen, and Timothy Hospedales. 2019. Tucker: Tensor factorization for knowledge graph completion. In *Proc. of Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing*, pages 5185–5194.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proc. of ACM International Conference on Management of Data*, pages 1247–1250.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Proc. of Annual Conference on Neural Information Processing Systems*, pages 1–9.

Zongsheng Cao, Qianqian Xu, Zhiyong Yang, Xiaochun Cao, and Qingming Huang. 2021. Dual quaternion knowledge graph embeddings. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 35, pages 6894–6902.

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

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 32, pages 1811–1818.

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

Takuma Ebisu and Ryutaro Ichise. 2018. Toruse: Knowledge graph embedding on a lie group. In *Proc. of AAAI Conference on Artificial Intelligence*.

Chang Gao, Chengjie Sun, Lili Shan, Lei Lin, and Mingjiang Wang. 2020. Rotate3d: Representing relations as rotations in three-dimensional space for knowledge graph embedding. In *Proc. of ACM International Conference on Information & Knowledge Management*, pages 385–394.

William Rowan Hamilton. 1844. Lxxviii. on quaternions; or on a new system of imaginaries in algebra: To the editors of the philosophical magazine and journal. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 25(169):489–495.

Guoliang Ji, Kang Liu, Shizhu He, and Jun Zhao. 2016. Knowledge graph completion with adaptive sparse transfer matrix. In *Proc. of AAAI Conference on Artificial Intelligence*, pages 985–991.

Seyed Mehran Kazemi and David Poole. 2018. Simple embedding for link prediction in knowledge graphs. *Proc. of Annual Conference on Neural Information Processing Systems*, 31.

Denis Krompaß, Stephan Baier, and Volker Tresp. 2015. Type-constrained representation learning in knowledge graphs. In *Proc. of International semantic web conference*, pages 640–655.

Timothée Lacroix, Nicolas Usunier, and Guillaume Obozinski. 2018. Canonical tensor decomposition for knowledge base completion. In *Proc. of International Conference on Machine Learning*, pages 2863–2872.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. 2015. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic web*, 6(2):167–195.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Proc.*

of AAAI Conference on Artificial Intelligence*, volume 29, pages 2181–2187.

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

Dai Quoc Nguyen, Thanh Vu, Tu Dinh Nguyen, and Dinh Phung. 2022. Quatre: Relation-aware quaternions for knowledge graph embeddings. In *Companion Proc. of International World Wide Web Conferences 2022*.

Maximilian Nickel, Lorenzo Rosasco, and Tomaso Poggio. 2016. Holographic embeddings of knowledge graphs. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 30, pages 1955–1961.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A three-way model for collective learning on multi-relational data. In *Proc. of International Conference on Machine Learning*, volume 11, pages 809–816.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *Proc. of European Semantic Web Conference*, pages 593–607.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. In *Proc. of International Conference on Learning Representations*, pages 1–18.

Kristina Toutanova and Danqi Chen. 2015. Observed versus latent features for knowledge base and text inference. In *Proc. of 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 *Proc. of International Conference on Machine Learning*, volume 48, pages 2071–2080.

Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, Nilesh Agrawal, and Partha Talukdar. 2020. Interacte: Improving convolution-based knowledge graph embeddings by increasing feature interactions. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 34, pages 3009–3016.

Kai Wang, Yu Liu, Xiujuan Xu, and Dan Lin. 2018. Knowledge graph embedding with entity neighbors and deep memory network. *arXiv preprint arXiv:1808.03752*.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 28, pages 1112–1119.

Ruobing Xie, Zhiyuan Liu, Maosong Sun, et al. 2016. Representation learning of knowledge graphs with hierarchical types. In *Proc. of International Joint Conference on Artificial Intelligence*, volume 2016, pages 2965–2971.

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 *Proc. of International Conference on Learning Representations*, pages 1–12.

Shuai Zhang, Yi Tay, Lina Yao, and Qi Liu. 2019. Quaternion knowledge graph embeddings. In *Proc. of Annual Conference on Neural Information Processing Systems*, volume 32, pages 2735–2745.

Zhanqiu Zhang, Jianyu Cai, Yongdong Zhang, and Jie Wang. 2020. Learning hierarchy-aware knowledge graph embeddings for link prediction. In *Proc. of AAAI Conference on Artificial Intelligence*, pages 3065–3072.

Zhao Zhang, Fuzhen Zhuang, Meng Qu, Fen Lin, and Qing He. 2018. Knowledge graph embedding with hierarchical relation structure. In *Proc. of Conference on Empirical Methods in Natural Language Processing*, pages 3198–3207.

A Appendix

A.1 Definitions of Different Relation Patterns

Definition 1 *Relations r_i are multiple if $\forall i \in 0, ..., m$, (h, r, t) can hold in knowledge graphs simultaneously. A clause with such form is a multiple relations pattern.*

Definition 2 *A relation r is symmetric (antisymmetry) if $\forall x, y$*

$$r(x, y) \Rightarrow r(y, x) \; (r(x, y) \Rightarrow \neg r(y, x)).$$

A clause with such form is a symmetry (antisymmetry) pattern.

Definition 3 *Relation r_1 is inverse to relation r_2 if $\forall x, y$*

$$r_2(x, y) \Rightarrow r_1(y, x).$$

A clause with such form is an inversion pattern.

Definition 4 *Relation r_1 is composed of relation r_2 and relation r_3 if $\forall x, y, z$*

$$r_2(x, y) \wedge r_3(y, z) \Rightarrow r_1(x, z).$$

A clause with such form is a composition pattern.

Definition 5 *For each relation r, we compute average number of tails per head (tphr) and average number of heads per tail (hptr). If tphr < 1.5 and hptr < 1.5, r is treated as 1-to-1; if tphr > 1.5 and hptr > 1.5, r is treated as a N-to-N; if tphr > 1.5 and hptr < 1.5, r is treated as 1-to-N. Clauses with such form are complex relations.*

A.2 Proof of Lemma 1

Proof of symmetry pattern. When $\theta = [0°, 180°, 360°]$, HRQE can represent a symmetric relationship, then $W_r^{\triangleleft} = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}(q\mathbf{i} + u\mathbf{j} + v\mathbf{k})$, we need to prove:

$$W_r^{\triangleleft} \otimes Q_h \otimes \bar{W}_r^{\triangleleft} \cdot Q_t = W_r^{\triangleleft} \otimes Q_t \otimes \bar{W}_r^{\triangleleft} \cdot Q_h \tag{9}$$

For $\theta = [0°, 360°]$, we have $W_r^{\triangleleft} = \pm 1$. Firstly, we expand the left term:

$$\begin{aligned}
&W_r^{\triangleleft} \otimes Q_h \otimes \bar{W}_r^{\triangleleft} \cdot Q_t \\
&= Q_h \cdot Q_t \\
&= \langle x_h, x_t \rangle + \langle y_h, y_t \rangle + \langle z_h, z_t \rangle.
\end{aligned} \tag{10}$$

We then expand the right term:

$$\begin{aligned}
&W_r^{\triangleleft} \otimes Q_t \otimes \bar{W}_r^{\triangleleft} \cdot Q_h \\
&= Q_t \cdot Q_h \\
&= \langle x_h, x_t \rangle + \langle y_h, y_t \rangle + \langle z_h, z_t \rangle.
\end{aligned} \tag{11}$$

We can easily see that those two terms are equal.

For $\theta = [180°]$, we have $W_r^{\triangleleft} = q\mathbf{i} + u\mathbf{j} + v\mathbf{k}$. Firstly, we expand the left term:

$$\begin{aligned}
&W_r^{\triangleleft} \otimes Q_h \otimes \bar{W}_r^{\triangleleft} \cdot Q_t \\
&= [((qq - uu - vv)x_h + 2(qu)y_h + 2(qv)z_h)\mathbf{i} \\
&+ (2(qu)x_h + (qq + uu - vv)y_h + 2(uv)z_h)\mathbf{j} \\
&+ (2(qv)x_h + 2(uv)y_h + (qq - uu + vv)z_h)\mathbf{k}] \\
&\cdot (x_t\mathbf{i} + y_t\mathbf{j} + z_t\mathbf{k}) \\
&= \langle x_h, (qq - uu - vv), x_t \rangle + \langle x_h, 2(qu), y_t \rangle \\
&+ \langle x_h, 2(qv), z_t \rangle + \langle y_h, 2(qu), x_t \rangle \\
&+ \langle y_h, (qq + uu - vv), y_t \rangle + \langle y_h, 2(uv), z_t \rangle \\
&+ \langle z_h, 2(qv), x_t \rangle + \langle z_h, 2(uv), y_t \rangle \\
&+ \langle z_h, (qq - uu + vv), z_t \rangle.
\end{aligned} \tag{12}$$

We then expand the right term:

$$\begin{aligned}
&W_r^{\triangleleft} \otimes Q_t \otimes \bar{W}_r^{\triangleleft} \cdot Q_h \\
&= [((qq - uu - vv)x_t + 2(qu)y_t + 2(qv)z_t)\mathbf{i} \\
&+ (2(qu)x_t + (qq + uu - vv)y_t + 2(uv)z_t)\mathbf{j} \\
&+ (2(qv)x_t + 2(uv)y_t + (qq - uu + vv)z_t)\mathbf{k}] \\
&\cdot (x_h\mathbf{i} + y_h\mathbf{j} + z_h\mathbf{k}) \\
&= \langle x_t, (qq - uu - vv), x_h \rangle + \langle x_t, 2(qu), y_h \rangle \\
&+ \langle x_t, 2(qv), z_h \rangle + \langle y_t, 2(qu), x_h \rangle \\
&+ \langle y_t, (qq + uu - vv), y_h \rangle + \langle y_t, 2(uv), z_h \rangle \\
&+ \langle z_t, 2(qv), x_h \rangle + \langle z_t, 2(uv), y_h \rangle \\
&+ \langle z_t, (qq - uu + vv), z_h \rangle.
\end{aligned} \tag{13}$$

We can easily see that those two terms are equal.

Proof of antisymmetry pattern. In order to prove the antisymmetry pattern, we need to prove the following inequality when $\theta \neq [0°, 180°, 270°, 360°]$:

$$W_r^{\triangleleft} \otimes Q_h \otimes \bar{W}_r^{\triangleleft} \cdot Q_t \neq W_r^{\triangleleft} \otimes Q_t \otimes \bar{W}_r^{\triangleleft} \cdot Q_h \tag{14}$$

Firstly, we expand the left term:

$$\begin{aligned}
&W_r^{\triangleleft} \otimes Q_h \otimes \bar{W}_r^{\triangleleft} \cdot Q_t \\
&= [((pp + qq - uu - vv)x_h + 2(-pv + qu)y_h \\
&+ 2(pu + qv)z_h)\mathbf{i} + (2(pv + qu)x_h \\
&+ (pp - qq + uu - vv)y_h + 2(-pq + uv)z_h)\mathbf{j} \\
&+ (2(-pu + qv)x_h + 2(pq + uv)y_h \\
&+ (pp - qq - uu + vv)z_h)\mathbf{k}] \\
&\cdot (x_t\mathbf{i} + y_t\mathbf{j} + z_t\mathbf{k}) \\
&= \langle x_h, (pp + qq - uu - vv), x_t \rangle \\
&+ \langle x_h, 2(pv + qu), y_t \rangle + \langle x_h, 2(-pu + qv), z_t \rangle \\
&+ \langle y_h, 2(-pv + qu), x_t \rangle \\
&+ \langle y_h, (pp - qq + uu - vv), y_t \rangle \\
&+ \langle y_h, 2(pq + uv), z_t \rangle + \langle z_h, 2(pu + qv), x_t \rangle \\
&+ \langle z_h, 2(-pq + uv), y_t \rangle \\
&+ \langle z_h, (pp - qq - uu + vv), z_t \rangle.
\end{aligned} \tag{15}$$

We then expand the right term:

Model	Score Function $f_r(Q_h, Q_t)$	Parameters	O_{time}
TransE (Bordes et al., 2013)	$-\|(Q_h + W_r) - Q_t\|_{1/2}$	$Q_h, W_r, Q_t \in \mathbb{R}^k$	$O(k)$
Hole (Nickel et al., 2016)	$\langle W_r, Q_h \star Q_t \rangle$	$Q_h, W_r, Q_t \in \mathbb{R}^k$	$O(k\log(k))$
DistMult (Yang et al., 2015)	$\langle W_r, Q_h, Q_t \rangle$	$Q_h, W_r, Q_t \in \mathbb{R}^k$	$O(k)$
ComplEx (Trouillon et al., 2016)	$\mathrm{Re}\left(\langle W_r, Q_h, \bar{Q}_t \rangle\right)$	$Q_h, W_r, Q_t \in \mathbb{C}^k$	$O(k)$
RotatE (Sun et al., 2019)	$-\|Q_h \circ W_r - Q_t\|_2$	$Q_h, W_r, Q_t \in \mathbb{C}^k, \|W_{ri}\| = 1$	$O(k)$
Rotate3D (Gao et al., 2020)	$-\|Q_h \odot W_r \times B_r - Q_t\|_2$	$Q_h, W_r, Q_t \in \mathbb{H}^k, B_r \in \mathbb{R}^k, \|W_{ri}\| = 1$	$O(k)$
QuatE (Zhang et al., 2019)	$Q_h \otimes W_r^{\triangleleft} \cdot Q_t$	$Q_h, W_r, Q_t \in \mathbb{H}^k$	$O(k)$
DualE (Cao et al., 2021)	$\left\langle Q_h \underline{\otimes} W_r^{\diamond}, Q_t \right\rangle$	$Q_h, W_r, Q_t \in \mathbb{H}_d^k$	$O(k)$
HAKE (Zhang et al., 2020)	$-\|Q_{hm} \circ W_m - Q_{tm}\|_2$ $-\lambda\|\sin((Q_{hp} + W_p - Q_{tp})/2)\|_1$	$Q_{hm}, Q_{tm} \in \mathbb{R}^k, W_m \in \mathbb{R}_+^k$ $Q_{hp}, W_p, Q_{tp} \in [0, \pi)^k, \lambda \in \mathbb{R}$	$O(k)$
RQE	$W_r^{\triangleleft} \otimes Q_h \otimes \bar{W}_r^{\triangleleft} \cdot Q_t$	$W_r \in \mathbb{H}^k, Q_h, Q_t \in \mathbb{H}_p^k$	$O(k)$
HRQE	$W_r^{\triangleleft} \otimes Q_h \otimes \bar{W}_r^{\triangleleft} \cdot Q_t$ $-\lambda\|w_r\|Q_h\| - \|Q_t\|\|_1$	$W_r \in \mathbb{H}^k, Q_h, Q_t \in \mathbb{H}_p^k$ $w_r \in \mathbb{R}^k, \lambda \in \mathbb{R}$	$O(k)$

Table 8: Scoring functions of state-of-the-art knowledge graph embedding models, along with their parameters, time complexity."★" denotes the circular correlation operation; "∘" denotes Hadmard (or element-wise) product; "⊗" denotes Hamilton product.

$$
\begin{aligned}
& W_r^{\triangleleft} \otimes Q_t \otimes \bar{W}_r^{\triangleleft} \cdot Q_h \\
& = [((pp + qq - uu - vv)x_t + 2(-pv + qu)y_t \\
& + 2(pu + qv)z_t)\mathbf{i} + (2(pv + qu)x_t \\
& + (pp - qq + uu - vv)y_t + 2(-pq + uv)z_t)\mathbf{j} \\
& + (2(-pu + qv)x_t + 2(pq + uv)y_t \\
& + (pp - qq - uu + vv)z_t)\mathbf{k}] \\
& \cdot (x_h\mathbf{i} + y_h\mathbf{j} + z_h\mathbf{k}) \\
& = \langle x_t, (pp + qq - uu - vv), x_h \rangle \\
& + \langle x_t, 2(pv + qu), y_h \rangle + \langle x_t, 2(-pu + qv), z_h \rangle \\
& + \langle y_t, 2(-pv + qu), x_h \rangle \\
& + \langle y_t, (pp - qq + uu - vv), y_h \rangle \\
& + \langle y_t, 2(pq + uv), z_h \rangle + \langle z_t, 2(pu + qv), x_h \rangle \\
& + \langle z_t, 2(-pq + uv), y_h \rangle \\
& + \langle z_t, (pp - qq - uu + vv), z_h \rangle.
\end{aligned}
\tag{16}
$$

We can easily see that those two terms are not equal as the signs for some terms are not the same.

A.3 Proof of Lemma 2

Proof of inversion pattern. To prove the inversion pattern, we need to prove that:

$$
W_r^{\triangleleft} \otimes Q_h \otimes \bar{W}_r^{\triangleleft} \cdot Q_t = \bar{W}_r^{\triangleleft} \otimes Q_t \otimes W_r^{\triangleleft} \cdot Q_h \tag{17}
$$

We expand the right term:

$$
\begin{aligned}
& \bar{W}_r^{\triangleleft} \otimes Q_t \otimes W_r^{\triangleleft} \cdot Q_h \\
& = [((pp + qq - uu - vv)x_t + 2(pv + qu)y_t \\
& + 2(-pu + qv)z_t)\mathbf{i} + (2(-pv + qu)x_t \\
& + (pp - qq + uu - vv)y_t + 2(pq + uv)z_t)\mathbf{j} \\
& + (2(pu + qv)x_t + 2(-pq + uv)y_t \\
& + (pp - qq - uu + vv)z_t)\mathbf{k}] \\
& \cdot (x_h\mathbf{i} + y_h\mathbf{j} + z_h\mathbf{k}) \\
& = \langle x_t, (pp + qq - uu - vv), x_h \rangle \\
& + \langle x_t, 2(-pv + qu), y_h \rangle + \langle x_t, 2(pu + qv), z_h \rangle \\
& + \langle y_t, 2(pv + qu), x_h \rangle \\
& + \langle y_t, (pp - qq + uu - vv), y_h \rangle \\
& + \langle y_t, 2(-pq + uv), z_h \rangle + \langle z_t, 2(-pu + qv), x_h \rangle \\
& + \langle z_t, 2(pq + uv), y_h \rangle \\
& + \langle z_t, (pp - qq - uu + vv), z_h \rangle.
\end{aligned}
\tag{18}
$$

We can easily check the equality of these two terms.

A.4 Proof of Lemma 3

Proof of composition relation. For composition relations we can get that:

$$
\begin{aligned}
& W_{r3}^{\triangleleft} \otimes (W_{r2}^{\triangleleft} \otimes Q_h \otimes \bar{W}_{r2}^{\triangleleft}) \otimes \bar{W}_{r3}^{\triangleleft} \cdot Q_t \\
& = (W_{r3}^{\triangleleft} \otimes W_{r2}^{\triangleleft}) \otimes Q_h \otimes (\bar{W}_{r2}^{\triangleleft} \otimes \bar{W}_{r3}^{\triangleleft}) \cdot Q_t \\
& = W_{r1}^{\triangleleft} \otimes Q_h \otimes \bar{W}_{r1}^{\triangleleft} \cdot Q_t.
\end{aligned}
\tag{19}
$$

A.5 Summary of Several Popular Knowledge Graph Embedding Models

Table 8 summarizes several popular knowledge graph embedding models, including scoring functions, parameters, and time complexities. TransE, HolE, and DistMult use Euclidean embeddings, while ComplEx and RotatE operate in the complex space. QuatE, DualE (dual quaternion) and our models operate in the quaternion space. HAKE and our model HRQE can learn hierarchy-aware in knowledge graphs.

A.6 Parameter Settings

We list the best hyperparameter settings of RQE and HRQE w.r.t. the validation dataset on several benchmarks in Table 9 and Table 10.

Dataset	nB	k	λ_1	λ_2	neg
WN18	10	300	0.03	0.0	10
FB15K	100	400	0.05	0.0	10
WN18RR	10	300	0.3	0.3	2
FB15K237	100	500	0.5	0.01	10

Table 9: Hyperparameters for RQE

Dataset	nB	k	λ_1	λ_2	neg
WN18	10	300	0.05	0.01	10
FB15K	100	500	0.03	0.0	10
WN18RR	10	300	0.3	0.01	1
FB15K237	100	500	0.5	0.01	10

Table 10: Hyperparameters for HRQE