5★ Knowledge Graph Embeddings with Projective Transformations

Mojtaba Nayyeri[1,2], **Sahar Vahdati**[2], **Can Aykul**[1], **Jens Lehmann**[1,3]
[1]Smart Data Analytics Group, University of Bonn, Germany
[2]Nature-Inspired Machine Intelligence-InfAI, Dresden, Germany
[3]Fraunhofer IAIS, Dresden, Germany
{nayyei,aykul,jens.lehmann}@cs.uni-bonn.de, vahdati@infai.org, jens.lehmann@iais.fraunhofer.de

Abstract

Performing link prediction using knowledge graph embedding models has become a popular approach for knowledge graph completion. Such models employ a transformation function that maps nodes via edges into a vector space in order to measure the likelihood of the links. While mapping the individual nodes, the structure of subgraphs is also transformed. Most of the embedding models designed in Euclidean geometry usually support a *single* transformation type – often translation or rotation, which is suitable for learning on graphs with small differences in neighboring subgraphs. However, multi-relational knowledge graphs often include multiple subgraph structures in a neighborhood (e.g. combinations of path and loop structures), which current embedding models do not capture well. To tackle this problem, we propose a novel KGE model (5★E) in projective geometry, which supports *multiple* simultaneous transformations – specifically inversion, reflection, translation, rotation, and homothety. The model has several favorable theoretical properties and subsumes the existing approaches. It outperforms them on most widely used link prediction benchmarks.

Introduction

Knowledge graphs (KGs) with their graph-based knowledge representation in the form of (head,relation,tail) triples, have become a leading technology of recent years in AI-based tasks including question answering, data integration, and recommender systems (Ji et al. 2020). However, KGs are incomplete and the performance of any algorithm consuming them is affected by this problem. Knowledge graph embeddings (KGEs) are a prominent approach used for KG completion by predicting missing links. Every KGE model uses a transformation function to map entities (nodes) of the graph through relations in a vector space to score the plausibility of triples via a score function. The performance of KGE models heavily relies on the design of their score function that in turn defines the type of transformation they support. Such transformations distinguish the extent to which a model is able to learn complex motifs and patterns formed by combinations of the nodes and edges.

Most of the existing KGEs have been designed in Euclidean geometry and usually support a single transformation type – often translation or rotation. This limits their ability in embedding KGs with complexities in subgraphs, especially when multiple structures exist in a neighborhood. An example of this situation is the presence of a path structure for a group of nodes close to a loop structure of another group in a KG (illustrated in Figure 1). The upper part of the figure shows examples of four different subgraphs containing combinations of path structures (a group of nodes connected via a relation) and loop structures (a group of nodes forming a loop via a relation). The lower part of the figure shows an exemplary visualisation of the embeddings of the entities depicted in the upper part of the figure. Let us focus on the left most example in the figure, i.e. the path to loop subgraph. In this example subgraph, a relation r_1 (e.g. *hypernym*) forms a path structure, a relation r_3 (e.g. *similar_to*) forms a loop structure and nodes in both structures are connected via a relation r_2. A loop in the graph presentation can be preserved as a circle in a vector space, and a path as a line. Existing KGE models, such as TransE, RotatE, ComplEx and QuatE partially capture those structures in the embedding space. The lower part of the figure shows the possible embeddings of the given subgraphs preserved by the existing models. Let the nodes in the path be $h_1, \ldots, h_6$, the nodes in the loop be $t_1, \ldots, t_6$ and $(h_i, r_2, t_i), i = 1, \ldots, 6$, be the triples connecting those structures. When we consider the TransE model specifically, the embeddings of the tails $t_1, \ldots, t_6$ cannot be transferred to the shape of a circle in the embedding space. This is because the following equations need to (approximately) hold according to the TransE score functions: $\mathbf{t}_1+\mathbf{r}_3 \approx \mathbf{t}_2$, $\mathbf{t}_2+\mathbf{r}_3 \approx \mathbf{t}_3$, $\ldots$, $\mathbf{t}_6+\mathbf{r}_3 \approx \mathbf{t}_1$. However, this results in $\mathbf{r}_3 = 0$. Therefore, $\mathbf{t}_1 = \mathbf{t}_2 = \ldots = \mathbf{t}_6$ i.e. all entities are embedded into the same point in the embedding space rather than a circle (with positive radius). Similar derivations apply to more recent and complex models. Those limitations are due to the limited set of transformations supported by those models, which do not go beyond translation, rotation and homothethy operations. Therefore, they cannot map line structures to circle structures and vice versa.

This type of limitation stems from the underlying geometry. While major existing models cover at most two transformation types, we propose a model based on projective geometry that provides a uniform way for *simultaneously* representing five *transformation types* namely translation, rotation, homothety, inversion, and reflection. The combination

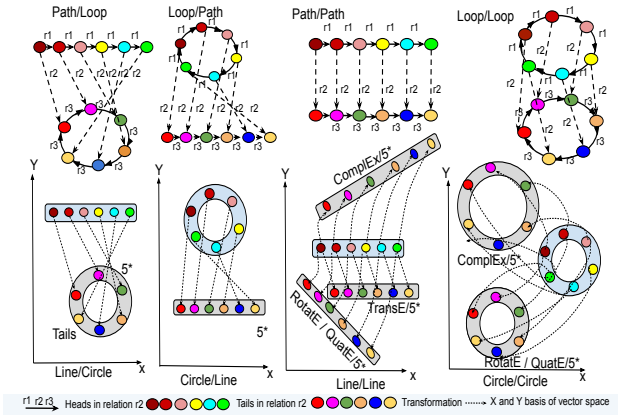


Figure 1: Graph and vector representation of path/loop.

of such transformation types results in various *transformation functions* (parabolic, circular, elliptic, hyperbolic, and loxodromic) subsumed by projective transformations. As a consequence, the embeddings can preserve the structure in the previous example and many other scenarios in which different structures exist in a neighbourhood of the KG.

Our core contribution is a new five-star embedding model, i.e. a model that simultaneously employs five transformation types and consequently can preserve various-shaped structures in the embedding space. The model subsumes several existing state-of-the-art KGE models , i.e. their score functions can be expressed as special cases of $5^\bigstar$E. Overall, our model, dubbed $5^\bigstar$E, is (a) capable of preserving a wider range of structures than existing models (including the path and loop combinations in Figure 1), (b) fully expressive (as defined in (Wang, Gemulla, and Li 2018)), (c) subsumes the KGE models DistMult, RotatE, pRotatE, TransE, and ComplEx (d) allows to learn composition, inverse, reflexive and symmetric relation patterns. Our evaluation shows that $5^\bigstar$E outperforms existing models on standard benchmarks.

Preliminaries and Background

Knowledge Graph Embeddings

A KG is a multi-relational directed graph $\mathcal{KG} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$ where $\mathcal{E}, \mathcal{R}$ are the set of nodes (entities) and edges (relations between entities) respectively. The set $\mathcal{T} = \{(h, r, t)\} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ contains all triples as *(head, relation, tail)*, e.g. (*smartPhone, hypernym, iPhone)*. In order to apply learning methods on KGs, a KGE learns vector representations of entities ($\mathcal{E}$) and relations ($\mathcal{R}$). A vector representation denoted by $(\mathbf{h}, \mathbf{r}, \mathbf{t})$ is learned by the model per triple (h, r, t), where $\mathbf{h}, \mathbf{t} \in \mathbb{V}^{d_e}$, $\mathbf{r} \in \mathbb{V}^{d_r}$ ($\mathbb{V}^d$ is a d-dimensional vector space). TransE (Bordes et al. 2013) considers $\mathbb{V} = \mathbb{R}$ while ComplEx (Trouillon et al. 2016) and RotatE use $\mathbb{V} = \mathbb{C}$ (complex space) and QuatE (Zhang et al. 2019) considers $\mathbb{V} = \mathbb{H}$ (quaternion space). In this paper, we choose a projective space to embed the graph i.e. $\mathbb{V} = \mathbb{CP}^1$ (a complex projective line which is introduced later). Most KGE models are defined via a relation-specific transformation function $g_r : \mathbb{V}^{d_e} \rightarrow \mathbb{V}^{d_e}$ which maps head entities to tail entities, i.e. $g_r(\mathbf{h}) = \mathbf{t}$. On top of such a transformation function, the score function $f : \mathbb{V}^{d_e} \times \mathbb{V}^{d_r} \times \mathbb{V}^{d_e} \rightarrow \mathbb{R}$ is defined to measure the plausibility for triples: $f(\mathbf{h}, \mathbf{r}, \mathbf{t}) = p(g_r(\mathbf{h}), \mathbf{t})$. Generally, the formulation of any score function can be either $p(g_r(\mathbf{h}), \mathbf{t}) = -\|g_r(\mathbf{h}) - \mathbf{t}\|$ or $p(g_r(\mathbf{h}), \mathbf{t}) = \langle g_r(\mathbf{h}), \mathbf{t} \rangle$.

Projective Geometry

Projective geometry uses *homogeneous coordinates* which represent N-dimensional coordinates with $N + 1$ numbers (i.e. use one additional parameter). For example, a point in 2D Cartesian coordinates, $[X, Y]$ becomes $[x, y, k]$ in homogeneous coordinates where $X = x/k, Y = y/k$ (in 1-dimensional real numbers, $[X]$ becomes $[x, y]$ where $X = x/y$). The key elements of projective geometry are as follows:

A **Projective Line** is a space in which a projective geometry is defined. A projective geometry requires a point at infinity to satisfy the axiom of "two parallel lines intersect in infinity". Therefore, an extended line $\mathbb{P}^1(\mathbb{K})$ ($\mathbb{K}$ is a real line) is realized with $\mathbb{K}$ and a point at infinity (which topologically is a circle). More concretely, the projective line is a set $\{[x, 1] \in \mathbb{P}^1(\mathbb{K}) | x \in \mathbb{K}\}$ with an additional member $[1 : 0]$ denoting the point at infinity. When $\mathbb{K} = \mathbb{C}$, the projective line is complex (complex projective line denoted by $\mathbb{CP}^1$).

The **Riemann Sphere** (illustrated in Figure 2) is an extended complex plane with a point at infinity. Precisely, it is built on a plane of complex numbers wrapped around a Sphere where poles denote 0 and ∞. In projective geometry, a complex projective line is a Riemann Sphere which used as a tool for projective transformations.

A **Projective Transformation** is the mapping of the Riemann Sphere to itself. Let $[x : y]$ be the homogeneous coordinates of a point in $\mathbb{CP}^1$. A projective transformation in $\mathbb{CP}^1$ is expressed by a matrix multiplication (Richter-Gebert 2011; Salomon 2007) as $\tau : \mathbb{CP}^1 \rightarrow \mathbb{CP}^1$:

$$\tau([x, y]) = \Im \begin{bmatrix} x \\ y \end{bmatrix}, \quad \Im = \begin{bmatrix} a & b \\ c & d \end{bmatrix}, \tag{1}$$

where the matrix $\Im$ must be invertible ($det(\Im) \neq 0$). By identifying $\mathbb{CP}^1$ with $\hat{\mathbb{C}} = \mathbb{C} \cup \{\infty\}$, a projective transformation is represented by a fractional expression through a sequence of homogenization, transformation, and dehomogenization as

$$x \rightarrow \begin{bmatrix} x \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} ax + b \\ cx + d \end{bmatrix} \rightarrow \frac{ax + b}{cx + d}, \tag{2}$$

where the mapping $\vartheta : \hat{\mathbb{C}} \rightarrow \hat{\mathbb{C}}$ is defined as

$$\vartheta(x) = \frac{ax + b}{cx + d}, \; ad - bc \neq 0. \tag{3}$$

The resulted mapping in Equation 3 describes all *Möbius* transformations.

The **Möbius Group** is the set of all Möbius transformations which is a projective linear group $PGL(2, \mathbb{C})$, i.e. the group of all 2×2 invertible matrices with the operation of matrix multiplication on a projective space. This group is the automorphism group $Aut(\hat{\mathbb{C}})$ of the Riemann Sphere $\hat{\mathbb{C}}$ or equivalently $\mathbb{CP}^1$.

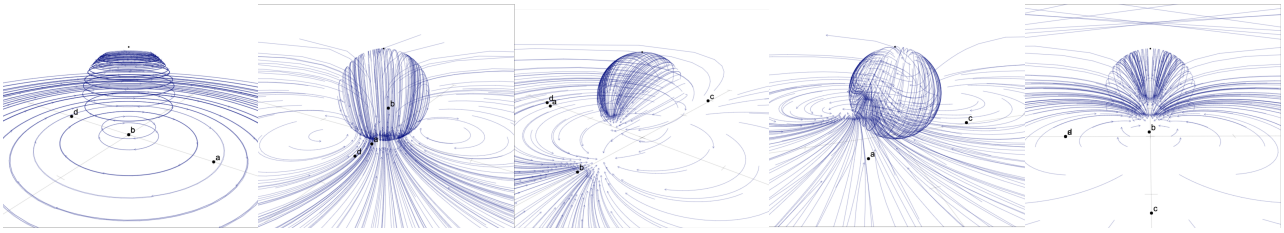


Figure 2: Transformation functions illustrated (Tim Hutton 2020) from left to right as circular, elliptic, hyperbolic, loxodromic, and parabolic. It shows a Riemann Sphere on a complex plane with one or two fix points for each function.

Variants of Möbius Transformations

Every Möbius transformation has at most two fixed points γ_1, γ_2 on the Riemann Sphere obtained by solving $\vartheta(\gamma) = \gamma$, (Richter-Gebert 2011) which gives $\gamma_{1,2} = \frac{(a-d)\pm\sqrt{\Delta}}{2c}$. Depending on the number of fixed points, Möbius transformations form parabolic or circular (one fixed point), elliptic as well as hyperbolic, and loxodromic (two fixed points) transformation functions illustrated in Figure 2. A Möbius transformation is performed on a grid by (a) a stereographic projection from complex plane to Riemann Sphere, (b) moving the Sphere, (c) stereographic projection from Sphere to plane. Each transformation has a constant $k = e^{\alpha+i\beta}$ which determines *sparsity/density* of the transformation. β is an expansion factor indicating the extent to which the fixed point γ_1 is repulsive (γ_2 is attractive). α is a rotation factor, determining the degree to which a transformation rotates the plane counter-clockwise around γ_1 (clockwise around γ_2).

Related Work

KGE models can be classified according to their embedding space. We discuss KGEs in Euclidean space and then describe related work for other geometries.

Euclidean Embedding Models A large number of KGE models such as TransE (Bordes et al. 2013) and its variants (Ji et al. 2015; Lin et al. 2015; Wang et al. 2014) as well as RotatE (Sun et al. 2019) are designed using translational or rotational (Hadamard product) score functions in Euclidean space. The score and loss functions of these models optimize the embedding vectors in a way that maximise the plausibility of triples, which is measured by the distance between rotated/translated head and tail vectors. Some embedding models such as DisMult (Yang et al. 2015), ComplEx (Trouillon et al. 2016), QuatE (Zhang et al. 2019), and RESCAL (Nickel, Tresp, and Kriegel 2011), including our proposed model, are designed based on element-wise multiplication of transformed head and tail. In this case, the plausibility of triples is measured based on the angle of transformed head and tail. A third category of KGE models are those designed on top of Neural Networks (NN) as score function such as ConvE (Dettmers et al. 2018) and NTN (Socher et al. 2013).

Non-Euclidean Embedding Models The aforementioned KGE models are limited to Euclidean space, which limits their ability to embed complex structures. Some recent efforts (Weber and Nickel 2018; Chami et al. 2020) investigated other spaces for embeddings of structures - often simpler structures than KGs. For example, the hyperbolic space has been extensively studied in scale-free networks. In recent work, learning continuous hierarchies from unstructured similarity scores using the Lorentz model was investigated (Nickel and Kiela 2018). In (Balazevic, Allen, and Hospedales 2019a), an embedding model dubbed MuRP is proposed that embeds multi-relational KGs on a Poincaré ball (Ji et al. 2016). MuRP only focuses on resolving the problem of embedding on KGs with multiple simultaneous hierarchies. Overall, while the advantages of projective geometry are eminent in a wide variety of application domains, including computer vision and robotics, to our knowledge no investigation has focused on it within the context of knowledge graph embeddings.

Method

Our method 5★E inherits the five main pillars of projective transformation, namely translation, rotation, homothety, inversion and reflection. The transformations are performed in the following steps: (1) *element-wise stereographic projection* to map the head entity from a complex plane into a point on a Riemann Sphere; (2) *relation-specific transformation* to move the Riemann Sphere into a new position and/or direction; (3) *stereographic projection* to project the mapped head from the Riemann Sphere to a complex plane (1-3 in Equations 4 and 5), (4) *selection of complex inner product* between the transformed head and the tail (Equation 6).

Model Formulation

Embedding on a Complex Projective Line Given a triple (h, r, t), the head and tail entities $h, t \in \mathcal{E}$ are embedded into a d dimensional complex projective line i.e. $\mathbf{h}, \mathbf{t} \in \mathbb{CP}^d$. A relation $r \in \mathcal{R}$ is embedded into a d dimensional vector $\mathbf{r}$ where each element is a 2×2 matrix. $\mathbf{r}$ contains four complex vectors $\mathbf{r_a}, \mathbf{r_b}, \mathbf{r_c}$ and $\mathbf{r_d} \in \mathbb{C}^d$. With $\mathbf{r}_{ai}, \mathbf{r}_{bi}, \mathbf{r}_{ci}, \mathbf{r}_{di}, \mathbf{h}_i, \mathbf{t}_i$, we refer to the ith element of $\mathbf{r}_a, \mathbf{r}_b, \mathbf{r}_c, \mathbf{r}_d, \mathbf{h}, \mathbf{t}$ respectively.

Relation-specific Transformation Based on preliminaries a projective transformation on a complex projective line has an equivalent transformation on the Riemann Sphere. Therefore, we use both of these perspectives in our model formulation.

Möbius Representation of Transformation: We use a relation-specific Möbius transformation to map the head entity ($\mathbf{h}_{ri}$) from a source to a target complex plane ($\hat{\mathbb{C}}$). The

transformation is performed using stereographic projection and transformation (ϑ) on/from the Riemann Sphere. To do so, we compute $\mathbf{h}_{ri}$ to specify the element-wise transformation:

$$\mathbf{h}_{ri} = g_{ri}(\mathbf{h}_i) = \vartheta(\mathbf{h}_i, \mathbf{r}_i) = \frac{\mathbf{r}_{ai}\mathbf{h}_i + \mathbf{r}_{bi}}{\mathbf{r}_{ci}\mathbf{h}_i + \mathbf{r}_{di}}, \quad \mathbf{r}_{ai}\mathbf{r}_{di} - \mathbf{r}_{bi}\mathbf{r}_{ci} \neq 0, i = 1, \ldots, d. \tag{4}$$

This results in the relation-specific transformed head entity $\mathbf{h}_r = [\mathbf{h}_{r1}, \ldots, \mathbf{h}_{rd}]$.

Projective Representation of Transformation: Using homogeneous coordinates, we can also represent Equation 4 as a projective transformation:

$$\mathbf{h}_{ri} \doteq [g_r(\mathbf{h}_i), 1]^T = \Im_{ri}[\mathbf{h}_i, 1]^T, i = 1, \ldots, d, \tag{5}$$

where $\doteq$ shows dehomogenization, $\Im_{ri} = \begin{bmatrix} \mathbf{r}_{ai} & \mathbf{r}_{bi} \\ \mathbf{r}_{ci} & \mathbf{r}_{di} \end{bmatrix}$ and $\det \Im_{ri} \neq 0$ i.e. $\Im_{ri}s$ are invertible. The matrix representation of Equation 5 is $\mathbf{h}_r = \mathbf{R}_\mathbf{r}[\mathbf{h} : \mathbf{1}]$, where $\mathbf{R}_\mathbf{r} = diag(\Im_{r1} \ldots, \Im_{rd})$ and $\mathbf{1}$ is a vector with all the elements being 1.

Score Function The correctness of triples in a KG is the similarity $\langle \mathbf{h}_r, \mathbf{t} \rangle$ between the relation-specific transformed head $\mathbf{h}_r$ and tail $\mathbf{t}$. The model aims to minimize the angle between $\mathbf{h}_r$ and tail $\mathbf{t}$, i.e. their product ($\langle \mathbf{h}_r, \mathbf{t} \rangle$) is maximized for positive triples. For sampled negative triples, it is conversely minimized. Overall, the score function for 5★E is

$$f(h, r, t) = Re(\langle \mathbf{h}_r, \bar{\mathbf{t}} \rangle), \tag{6}$$

where $Re(x)$ is the real part of the complex number x.

Theoretical Analysis

We first show that 5★E covers the five transformations. We then discuss the capability of 5★E in preserving graph structures. We also prove 5★E is fully expressive and subsumes various state-of-the-art KGE models.

Möbius – Composition of Transformations The Möbius transformation in Equation 4 is a composition of a series of five subsequent transformations ϑ_1, ϑ_2(two transformations in one), ϑ_3 and ϑ_4 as shown in (Kisil 2012): $\mathbf{h}_{ri} = \vartheta(\mathbf{h}_i, \mathbf{r}_i) = \vartheta_4 \circ \vartheta_3 \circ \vartheta_2 \circ \vartheta_1(\mathbf{h}_i, \mathbf{r}_i)$, where $\vartheta_1(\mathbf{x}, \mathbf{r}_i) = \mathbf{x} + \frac{\mathbf{r}_{di}}{\mathbf{r}_{ci}}$ (translation by $\frac{\mathbf{r}_{di}}{\mathbf{r}_{ci}}$), $\vartheta_2(\mathbf{x}) = \frac{1}{\mathbf{x}}$ (inversion and reflection w.r.t. real axis), $\vartheta_3(\mathbf{x}, \mathbf{r}_i) = \frac{\mathbf{r}_{bi}\mathbf{r}_{ci} - \mathbf{r}_{ai}\mathbf{r}_{di}}{\mathbf{r}_{ci}^2}\mathbf{x}$ (homothety and rotation) and $\vartheta_4(\mathbf{x}, \mathbf{r}_i) = \mathbf{x} + \frac{\mathbf{r}_{ai}}{\mathbf{r}_{ci}}$ (translation by $\frac{\mathbf{r}_{ai}}{\mathbf{r}_{ci}}$). This shows that 5★E is capable of performing 5 transformations simultaneously.

Capturing Structures in a Neighborhood 5★E inherits various important properties of projective transformation as well as Möbius transformations. Because the projective linear group $PGL(2, \mathbb{C})$ is isomorphic to the Möbius group, i.e., $PGL(2, \mathbb{C}) \cong Aut(\hat{\mathbb{C}})$ (Kisil 2012), the properties which are mentioned for Equation 5 are also valid for Equation 4. We investigate the inherited properties of 5★E on *clustering similar nodes of a neighborhood* and *Capturing Sub-graph Structures*.

Clustering. The similarity of nodes in a KG is local, i.e. nodes within a close neighborhood are more likely to be semantically similar (Faerman et al. 2018; Hamilton, Ying, and Leskovec 2017) than nodes at a higher distance. A projective transformation is a bijective conformal mapping, i.e. it preserves angle locally but not necessarily the length. It also preserves orientation after mapping (Kisil 2012). Therefore, 5★E is capable of capturing similarity by preserving angle locally via a relation-specific transformation.

Furthermore, the map $\pi : GL(2, \mathbb{C}) \rightarrow Aut(\hat{\mathbb{C}})$ is a group homomorphism, where $GL(2, \mathbb{C})$ is a generalized linear group, which transfers the matrix $\Im$ into a Möbius transformation ϑ. If $\det \Im = 1$, then $\pi : SL(2, \mathbb{C}) \rightarrow Aut(\hat{\mathbb{C}})$ becomes limited to only perform a mapping from the special linear group $SL(2, \mathbb{C})$ to a Möbius group that preserves volume and orientation.

In the context of KGs, after a relation-specific transformation (Equation 5 or equivalently Equation 4) of nodes in the head position to nodes in tail position, the relative distance of nodes can be preserved. From this ability, we expect that 5★E is able to propagate the structural similarity from one group of nodes to another.

Capturing Sub-graph Structures. Going beyond $SL(2, \mathbb{C})$ by changing the determinant to $\det \Im \neq 1$, the volume and orientation of the graph sub-structures are changed after transformation. Therefore, 5★E is more flexible than current KGEs as those are not able to change volume and orientation of subgraphs. This is visible in Figure 1 when the graph includes a group of nodes in a path structure besides another group of nodes with a loop structure. In the vector space, other KGEs encounter a problem in preserving this type of graph structure due to the limited transformation abilities (not supporting inversion and reflection), whereas they work fine for homogeneous structures (e.g. only lines or only circles). In contrast to this, 5★E is capable of transforming heterogeneous structures due to the characteristics of a projective transformation (Kisil 2012).

Subsumption of Other Models We show that 5★E subsumes other models and inherits their favorable characteristics in learning various graph patterns.

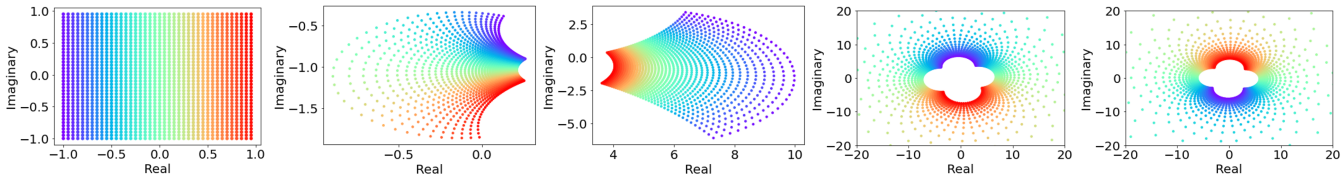
Definition 1. A model M_1 subsumes M_2 when any scoring over triples of a KG measured by model M_2 can also be obtained by M_1 (Wang, Gemulla, and Li 2018).

Proposition 1. *5★E with variants of its score function subsumes DistMult, pRotatE, RotatE, TransE and ComplEx. Specifically, 5★E subsumes DistMult, ComplEx and pRotatE with $f(h, r, t) = Re(\langle \mathbf{h}_r, \bar{\mathbf{t}} \rangle)$ and subsumes RotatE and TransE with score function $f(h, r, t) = -\|\mathbf{h}_r - \mathbf{t}\|$ (changed inner product to distance).*

Definition 2 (from (Kazemi and Poole 2018))**.** A model M is *fully expressive* if there exist assignments to the embeddings of the entities and relations, that accurately separate correct triples for any given ground truth.

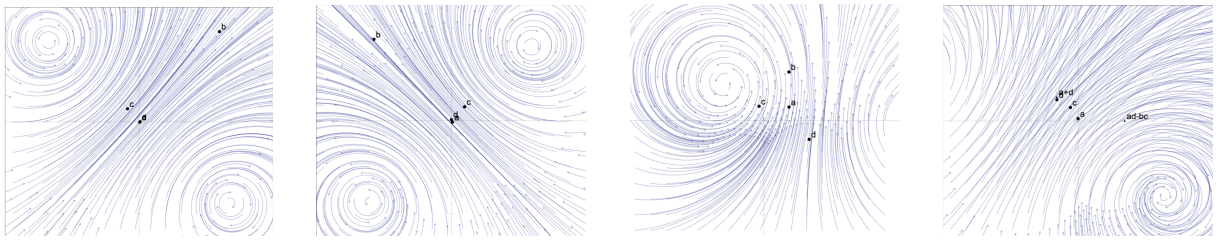
Corollary 1. *The 5★E model is fully expressive.*

Inference of Patterns For relations which exhibit patterns in the form of *premise* → *conclusion*, where *premise* can be a conjunction of several triples, a model is said to be able to infer those if the implication holds for the score function, i.e. if the score of all triples in the premise is positive then



(a) Original Grid (b) hasPart relation (c) partOf relation (d) hypernym (e) hyponym

Figure 3: Learned 5⋆E embeddings for a selected relations in WordNet. (b)-(e) show how the lines in (a) are transformed for a particular dimension (the 12th dimension in this case) of the embedding of the mentioned relations.



(a) 12-dim-hypernym (b) 12-dim-hyponym (c) 39-dim-haspart (d) 39-dim-partof

Figure 4: Embeddings for different relations usign the same dimension.

the score for the conclusion must be positive. 5⋆E is able to infer reflexive, symmetric, inverse relation patterns as well as composition.

Proposition 2. *Let $r_1, r_2, r_3 \in \mathcal{R}$ be relations and r_3 (e.g. UncleOf) be a composition of r_1(e.g. BrotherOf) and r_2(e.g. FatherOf). 5⋆E infers composition with $\Im_{r_1}\Im_{r_2} = \Im_{r_3}$.*

Proposition 3. *Let $r_1 \in \mathcal{R}$ be the inverse of $r_2 \in \mathcal{R}$. 5⋆E infers this pattern with $\Im_{r_1} = \Im_{r_2^{-1}}$.*

Proposition 4. *Let $r \in \mathcal{R}$ be symmetric. 5⋆E infers the symmetric pattern if $\Im_r = \Im_r^{-1}$.*

Proposition 5. *Let $r \in \mathcal{R}$ be a reflexive relation. In dimension d, 5⋆E infers reflexive patterns with $O(2^d)$ distinct representations of entities if the fixed points are non-identical.*

TransE only infers composition and inverse patterns. RotatE is capable of inferring more patterns but is not fully expressive. ComplEx infers these patterns and is fully expressive. However, it has less flexibility than our model in learning complex structures due to using only rotation and homothety. Therefore, it is only capable of preserving homogeneous structures (see Figure 5).

Experiments and Results

Experimental Setup Following the best practices of evaluations for embedding models, we consider the most-used metrics (Mean) Reciprocal Rank (MRR) and Hits@n (n = 1, 3, 10). We evaluated our model on four widely used benchmark datasets namely FB15k-237 (Toutanova and Chen 2015), WN18RR (Dettmers et al. 2018) , and NELL (four different versions as NELL-995-h25, NELL-995-h50, NELL-995-h75 and NELL-995-h100) (Xiong, Hoang, and Wang 2017; Balazevic, Allen, and Hospedales 2019a). The FB15k-237 and WN18RR datasets both include several relational patterns such as composition (e.g. $awardnominee/\ldots/nominatedfor$), symmetry (e.g $derivationally\_related\_form$ in WN18RR), and anti-symmetry (e.g has_part in WN18RR). The WN18RR dataset includes hierarchical relations such as $hypernym$ and has_part, which are typical examples for shaping a path structure, and relations such as $also\_see$, $similar\_to$, which are candidates for loop structures. The different variants of the NELL dataset include several relations that contain loops ($hassibling$, $competeswith$, $synonymfor$) as well as relations forming hierarchical paths ($subpartof$).

We compare the best performing models namely *TransE* (Bordes et al. 2013), *RotatE* (Sun et al. 2019), *TuckER* (Balazevic, Allen, and Hospedales 2019b), *ComplEx* (Trouillon et al. 2016), *QuatE* (Zhang et al. 2019), *MuRP* (Balazevic, Allen, and Hospedales 2019a), *ConvE* (Dettmers et al. 2018) and *SimplE* (Kazemi and Poole 2018). Our model is implemented in Pytorch[1] and the code is available online[2]. Similar to QuatE and ComplEx, we developed our model on top of a standard framework (Lacroix, Usunier, and Obozinski 2018), applied 1-N scoring loss with N3 regularization, and added reverse counterparts of each triple to the train set.

Results. The results of comparing 5⋆E to other models on FB15k-237 and WN18RR are shown in Table 1 ($d = 100$ and 500) and on NELL in Table 2 ($d = 100$ and 200). Our model outperforms all other models across all metrics on WN18RR, which is a dataset with many hierarchical relations as well as relations forming loops such as $similar - to$. Although

[1]https://pytorch.org/

[2]https://bit.ly/2NXplO1

Model	FB15k237				WN18RR			
	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10
TransE	0.29	-	-	0.47	0.23	-	-	0.50
RotatE	0.34	0.24	0.38	0.53	0.48	0.43	0.49	0.57
TuckEr	0.36	0.27	0.39	0.54	0.47	0.44	0.48	0.53
ComplEx	0.36	0.27	0.40	0.56.	0.49	0.44	0.50	0.58
QuatE	0.37	0.27	0.40	0.56	0.48	0.44	0.50	0.57
ConvE	0.33	0.24	0.36	0.50	0.43	0.40	0.44	0.52
MuRP	0.34	0.24	0.37	0.52	0.48	0.44	0.50	0.57
$5^\star$E d = 500	0.37	0.28	0.40	0.56	0.50	0.45	0.51	0.59
$5^\star$E d = 100	0.35	0.26	0.38	0.53	0.47	0.41	0.50	0.58

Table 1: Link prediction results on d FB15k-237, and WN18RR. The Results of TransE, QuatE, RotatE, and ConvE are taken from (Zhang et al. 2019), TuckER from (Balazevic, Allen, and Hospedales 2019b) and MuRP from (Balazevic, Allen, and Hospedales 2019a), and ComplEx has been experimented.

Model	NELL-995-h100				NELL-995-h75			
	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10
MuRE	0.36	0.27	0.40	0.53	0.36	0.27	0.40	0.53
MuRP	0.36	0.27	0.40	0.53	0.36	0.28	0.40	0.52
ComplEx	0.35	0.27	0.40	0.52	0.35	0.27	0.39	0.51
QuatE	0.35	0.26	0.40	0.53	0.36	0.27	0.41	0.52
$5^\star$E d = 200	0.37	0.28	0.42	0.54	0.37	0.28	0.41	0.53
$5^\star$E d = 100	0.36	0.28	0.40	0.53	0.36	0.27	0.39	0.53
Model	NELL-995-h50				NELL-995-h25			
	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10
MuRE	0.37	0.28	0.42	0.54	0.37	0.29	0.40	0.52
MuRP	0.37	0.28	0.42	0.54	0.36	0.28	0.40	0.51
ComplEx	0.37	0.29	0.41	0.52	0.37	0.30	0.40	0.51
QuatE	0.36	0.27	0.40	0.53	0.36	0.28	0.40	0.51
$5^\star$E d = 200	0.38	0.30	0.43	0.54	0.39	0.31	0.43	0.53
$5^\star$E d = 100	0.38	0.29	0.43	0.54	0.37	0.30	0.41	0.52

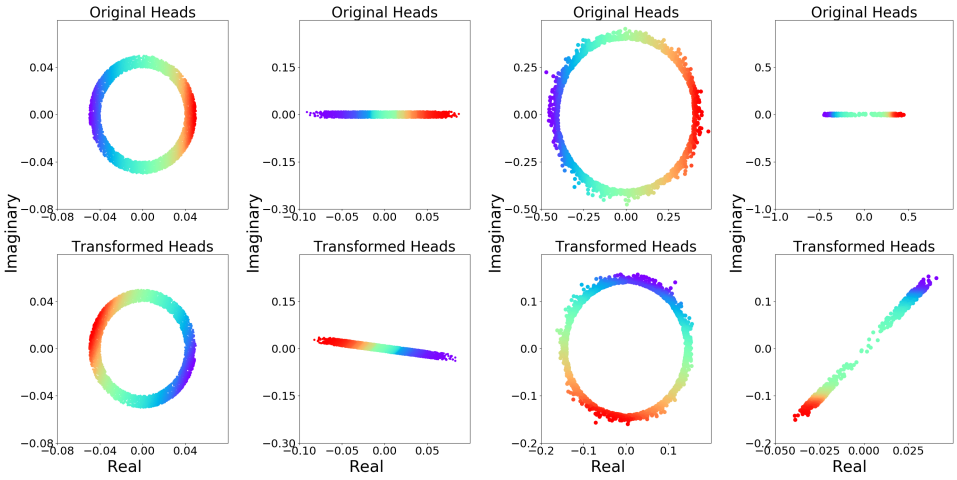
Table 2: Link prediction results on KGs with various percentages of hierarchical relations including NELL-995-h25 (25% hierarchical relation) and NELL-995-h50 (50%) as well as NELL-995-h75 (75%) and NELL-995-h100 (100%). The results for ComplEx, QuatE, and $5^\star$E are from own experiments - all others are taken from their original works.

MuRP is specifically designed for hierarchical data, $5^\star$E still achieves a better performance. Generally, we can observe that $5^\star$E obtains competitive results with a low dimension ($d = 100$) on WN18RR for the Hits@3 and Hits@10 metrics.

The evaluation shows that rotation-based models (RotatE, QuatE, and ComplEx) obtain state-of-the-art results on the FB15k-237 dataset (with fewer hierarchical paths than WN18RR). Our model, which covers rotation and transformation, obtains similar results to those models. On this dataset, there is no evident benefit of supporting further transformations. We additionally used different versions of the NELL dataset, which are specifically designed to have a particular percentage of hierarchical relations. $5^\star$E outperforms other models in all NELL dataset versions.

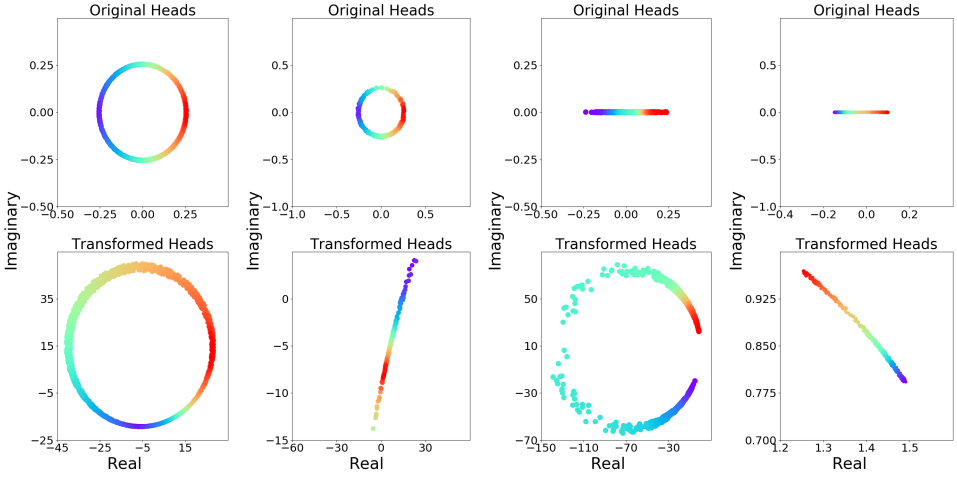
Overall, the competitive results of $5^\star$E show that additional transformations have a positive effect for the link prediction task. They also indicate that the additional transformations do not lead to over-fitting problems compared to single-transformation models (or at least the positive effects outweigh potential overfitting).

Learned Transformation Types Each relation in the KG is represented as d projective transformations in $5^\star$E (one projective transformation per dimension). Figure 3 shows the transformation types learned by $5^\star$E for the relations of WordNet. The original plain view of the grid is given in sub-graph (a) for comparisons of the changes after the transformations, and (b) to (e) show specific relations in WordNet. The mapping of the *lines* (same-color points) in the original grid to *circle or curve* in sub-graph (b)-(e) indicates an application of an inversion transformation for relation-specific transformations (*hasPart*, *partOf*, *hypernym* and *hyponym*). By comparing the direction of the lines with the same color (e.g., red) in the original grid and in all examples of the transformed grids, we can observe that the learned transformations cover rotation (*hypernym*, and *hyponym*). We can also interpret the results for the *hasPart* relation as counter-clockwise



(a) Circle-to-circle (b) line-to-line (c) Circle-to-circle (d) line-to-line

Figure 5: Types of transformations that RotatE (a,b) and ComplEx (c,d) learned on relation "hypernym" in WordNet. Each pair of images visualises one dimension of the relation embedding. The top images show the embeddings of head entities of this relation in the KG at this dimension. Each entity embedding is visualised as a colored dot. The bottom images show the results of applying the relation specific transformation (the entity colors are preserved).



(a) Circle-to-circle (b) circle-to-line (c) line-to-circle (d) line-to-line

Figure 6: Types of transformations that 5★E learned on relation "hypernym" in WordNet. Each pair of images visualises one dimension of the relation embedding. The top images show the embeddings of head entities of this relation in the KG at this dimension. Each entity embedding is visualised as a colored dot. The bottom images show the results of applying the relation specific transformation (the entity colors are preserved).

rotation and then reflection w.r.t. the real axis. In sub-graph (b), there is a movement in the real and imaginary axis of the grid towards down and slightly right for the *hasPart* relation, which represents translation. However, this is not the case for the *hypernym* relation. Semantically, the pairs (*hypernym*, *hyponym*) and (*hasPart partOf*) form inverse patterns (see Corollary 3). We see that the transformed grids of *hypernym* and *hyponym* are different only w.r.t. rotation. The scale is not changed, so the determinants of the two projective matrices are 1 (no homothety). For the *hasPart* and *partOf* grids, we can observe that the scale is changed, so the determinant of those two projection matrices should not be equal to one. This shows that both of those transformations cover *homothety*.

Moreover, each of the five transformation functions performed in Figure 2 are also learned by 5★E which confirms the flexibility of the model as well as diversity in density/sparsity of flows. Figure 4 shows the analysis on the example of *hyponym* and *hypernym* as well as *hasPart* and *partOf* relations which are mutually inverse of each other. Based on their inverse characteristic, we have $\Im_{Hypernym} = \Im^{-1}_{Hyponym}$. As the representing matrices ($\Im$) are normalized, their determinant is equal to one. Consequently,

(a) Clustering by ComplEx (b) Clustering by QuatE (c) Clustering by 5★E

Figure 7: Comparison of clustering results between ComplEx, QuatE, and 5★E (left to right). Each color shows a class of entities and each point is one instance entity of the corresponding class.

we have $tr(\Im_{Hypernym}) = tr(\bar{\Im}^{-1}_{Hyponym})$ when the learned transformation function is elliptic. We conclude that in our experiments for a pair of inverse relations, the learned transformation functions are in the same category (elliptic) for the i-th element of the relation embedding. In Figure 4, sub-figures (a) and (b) illustrate that the learned functions fall into the elliptic category for the same embedding dimension ($i = 12$) of *hyponym* and *hypernym*. The difference between the embeddings for this pair of inverse relations is their rotation. The same pattern is notable for the *hasPart* and *partOf* relations in sub-figures (c) and (d) for $i = 39$. Figure 5 shows the mapping of lines and circles by other KGEs. When observing each dimension of each relation, there was not a single case where a shape has been mapped to a different one, which empirically confirms our theoretical finding that existing models can only perform homogeneous transformations. In contrast, Figure 6 shows a relation-specific mapping of line to circle and circle to line performed by 5★E model.

Entity Clustering As mentioned in theoretical analysis, our model uses a bijective conformal mapping in the projective geometry which consequently preserves angle locally. In Figure 7, we provide an evaluation for the performance of the models in terms of clustering. More precisely, Figures 7a, 7b and 7c show the clustering of nodes in Freebase KG (Moon, Jones, and Samatova 2017) and illustrate comparisons to QuatE, ComplEx and 5★E. In this visualization, we can see that entities of the same type are closer in 5★E as compared to the competitors. Moreover, the distance between cluster centers in 5★E is higher than in the other models. Therefore, it is visible that 5★E provides a more suitable clustering for this dataset than other competitors.

Conclusion

In this paper, we introduce a new KGE model which operates on the complete set of projective transformations. We build the model on well researched generic mathematical foundations and showed that it subsumes other state-of-the-art embedding models. Furthermore, we prove that the model is fully expressive. By supporting a wider range of transformations than previous models, it can embed KGs with more complex structures and supports a wide range of relational patterns. We empirically studied and visualised the effects using the example of loop and path combinations. Our experimental evaluation on six benchmark datasets using established metrics shows that the model outperforms previous approaches of knowledge graph embedding models.

Acknowledgements

We acknowledge the support of the following projects: SPEAKER (BMWi FKZ 01MK20011A), JOSEPH (Fraunhofer Zukunftsstiftung), Cleopatra (GA 812997), the excellence clusters ML2R (BmBF FKZ 01 15 18038 A/B/C), MLwin (01IS18050), ScaDS.AI (IS18026A-F), TAILOR (EU GA 952215), and H2020-EU PLATOON (872592).

References

Balazevic, I.; Allen, C.; and Hospedales, T. 2019a. Multi-relational Poincaré graph embeddings. In *Advances in Neural Information Processing Systems*, 4465–4475.

Balazevic, I.; Allen, C.; and Hospedales, T. 2019b. TuckER: Tensor Factorization for Knowledge Graph Completion. In *EMNLP-IJCNLP Conference*, 5188–5197.

Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; and Yakhnenko, O. 2013. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, 2787–2795.

Chami, I.; Wolf, A.; Juan, D.-C.; Sala, F.; Ravi, S.; and Ré, C. 2020. Low-Dimensional Hyperbolic Knowledge Graph Embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 6901–6914.

Dettmers, T.; Minervini, P.; Stenetorp, P.; and Riedel, S. 2018. Convolutional 2d knowledge graph embeddings. In *Thirty-Second AAAI Conference*.

Faerman, E.; Borutta, F.; Fountoulakis, K.; and Mahoney, M. W. 2018. Lasagne: Locality and structure aware graph node embedding. In *International Conference on Web Intelligence (WI)*, 246–253. IEEE.

Hamilton, W. L.; Ying, R.; and Leskovec, J. 2017. Representation learning on graphs: Methods and applications. *arXiv preprint arXiv:1709.05584* .

Ji, G.; He, S.; Xu, L.; Liu, K.; and Zhao, J. 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)*, 687–696.

Ji, G.; Liu, K.; He, S.; and Zhao, J. 2016. Knowledge Graph Completion with Adaptive Sparse Transfer Matrix. 985–991.

Ji, S.; Pan, S.; Cambria, E.; Marttinen, P.; and Yu, P. S. 2020. A survey on knowledge graphs: Representation, acquisition and applications. *arXiv preprint arXiv:2002.00388* .

Kazemi, S. M.; and Poole, D. 2018. Simple embedding for link prediction in knowledge graphs. In *Advances in neural information processing systems*, 4284–4295.

Kisil, V. V. 2012. *Geometry of Möbius Transformations: Elliptic, Parabolic and Hyperbolic Actions of SL2 [real Number]*. World Scientific.

Lacroix, T.; Usunier, N.; and Obozinski, G. 2018. Canonical Tensor Decomposition for Knowledge Base Completion. In *International Conference on Machine Learning (ICML)*, 2863–2872.

Lin, Y.; Liu, Z.; Sun, M.; Liu, Y.; and Zhu, X. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Twenty-ninth AAAI conference on artificial intelligence*.

Moon, C.; Jones, P.; and Samatova, N. F. 2017. Learning entity type embeddings for knowledge graph completion. In *Proceedings of the 2017 ACM on conference on information and knowledge management*, 2215–2218.

Nickel, M.; and Kiela, D. 2018. Learning Continuous Hierarchies in the Lorentz Model of Hyperbolic Geometry. volume 80 of *Proceedings of Machine Learning Research*, 3779–3788. PMLR.

Nickel, M.; Tresp, V.; and Kriegel, H.-P. 2011. A Three-Way Model for Collective Learning on Multi-Relational Data. 11: 809–816.

Richter-Gebert, J. 2011. *Perspectives on projective geometry: A guided tour through real and complex geometry*. Springer Science & Business Media.

Salomon, D. 2007. *Transformations and projections in computer graphics*. Springer Science & Business Media.

Socher, R.; Chen, D.; Manning, C. D.; and Ng, A. 2013. Reasoning with neural tensor networks for knowledge base completion 926–934.

Sun, Z.; Deng, Z.-H.; Nie, J.-Y.; and Tang, J. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint arXiv:1902.10197* .

Tim Hutton. 2020. Mobius Transforms. https://github.com/timhutton/mobius-transforms. Online; accessed 15 December 2020.

Toutanova, K.; and Chen, D. 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*, 57–66.

Trouillon, T.; Welbl, J.; Riedel, S.; Gaussier, É.; and Bouchard, G. 2016. Complex embeddings for simple link prediction. In *International Conference on Machine Learning*, 2071–2080.

Wang, Y.; Gemulla, R.; and Li, H. 2018. On multi-relational link prediction with bilinear models. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Wang, Z.; Zhang, J.; Feng, J.; and Chen, Z. 2014. Knowledge graph embedding by translating on hyperplanes. In *Twenty-Eighth AAAI conference on artificial intelligence*.

Weber, M.; and Nickel, M. 2018. Curvature and Representation Learning: Identifying Embedding Spaces for Relational Data. *NeurIPS Relational Representation Learning* .

Xiong, W.; Hoang, T.; and Wang, W. Y. 2017. DeepPath: A Reinforcement Learning Method for Knowledge Graph Reasoning. In *EMNLP Conference*, 564–573.

Yang, B.; Yih, W.-t.; He, X.; Gao, J.; and Deng, L. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *Conference on Learning Representations (ICLR)*.

Zhang, S.; Tay, Y.; Yao, L.; and Liu, Q. 2019. Quaternion knowledge graph embeddings. In *Advances in Neural Information Processing Systems*, 2731–2741.