

Shrinking Embeddings for Hyper-Relational Knowledge Graphs

Bo Xiong¹, Mojtaba Nayyeri¹, Shirui Pan², Steffen Staab^{1,3}

¹University of Stuttgart, ²Griffith University, ³University of Southampton

Abstract

Link prediction on knowledge graphs (KGs) has been extensively studied on binary relational KGs, wherein each fact is represented by a triple. A significant amount of important knowledge, however, is represented by hyper-relational facts where each fact is composed of a primal triple and a set of qualifiers comprising a key-value pair that allows for expressing more complicated semantics. Although some recent works have proposed to embed hyper-relational KGs, these methods fail to capture essential inference patterns of hyper-relational facts such as qualifier monotonicity, qualifier implication, and qualifier mutual exclusion, limiting their generalization capability. To unlock this, we present *ShrinkE*, a geometric hyper-relational KG embedding method aiming to explicitly model these patterns. *ShrinkE* models the primal triple as a spatial-functional transformation from the head into a relation-specific box. Each qualifier “shrinks” the box to narrow down the possible answer set and, thus, realizes qualifier monotonicity. The spatial relationships between the qualifier boxes allow for modeling core inference patterns of qualifiers such as implication and mutual exclusion. Experimental results demonstrate *ShrinkE*’s superiority on three benchmarks of hyper-relational KGs.

1 Introduction

Link prediction on knowledge graphs (KGs) is a central problem for many KG-based applications (Zhang et al., 2016; Lukovnikov et al., 2017; Lu et al., 2023; Xiong et al., 2022b; Chen et al., 2022). Existing works (Sun et al., 2019; Bordes et al., 2013) have mostly studied link prediction on binary relational KGs, wherein each fact is represented by a triple, e.g., (*Einstein*, *educated_at*, *University of Zurich*). In many popular KGs such as Freebase (Bollacker et al., 2007), however, a lot of important knowledge is not only expressed in triple-shaped facts, but also via facts about facts, which

taken together are called hyper-relational facts. For example, (*Einstein*, *educated_at*, *University of Zurich*), $\{(major:physics), (degree:PhD)\}$ is a hyper-relational fact, where the primary triple (*Einstein*, *educated_at*, *University of Zurich*) is contextualized by a set of key-value pairs $\{(major:physics), (degree:PhD)\}$. Like much other related work, we follow the terminology established for Wikidata (Vrandečić and Krötzsch, 2014) and use the term *qualifiers* to refer to the key-value pairs.¹ The qualifiers play crucial roles in avoiding ambiguity issues. For instance, *Einstein* was *educated_at* several universities and the qualifiers for *degree* and *major* help distinguish them.

In order to predict links in hyper-relational KGs, pioneering works represent each hyper-relational fact as either an n -tuple in the form of $r(e_1, e_2, \dots, e_n)$ (Wen et al., 2016; Zhang et al., 2018; Fatemi et al., 2020; Liu et al., 2020; Abboud et al., 2020) or a set of key-value pairs in the form of $\{(k_i : v_i)\}_{i=1}^m$ (Guan et al., 2019, 2021; Liu et al., 2021a). However, these modelings lose key structure information and are incompatible with the RDF-star schema (Arndt et al., 2021) used by modern KGs, where both primal triples and qualifiers constitute the fundamental data structure. Recent works (Guan et al., 2020; Rosso et al., 2020) represent each hyper-relational fact as a primary triple coupled with a set of qualifiers that are compatible with RDF-star standards (Arndt et al., 2021). Link prediction is then achieved by modeling the validity of the primary triple and its compatibility with each annotated qualifier (Guan et al., 2020; Rosso et al., 2020). More complicated graph encoders and decoders (Galkin et al., 2020; Yu and Yang, 2021; Wang et al., 2021; Shomer et al., 2022) are proposed to further boost the performance. However, they require a relatively huge number of parameters

¹Synonyms include statement-level *metadata* in RDF-star (Arndt et al., 2021) and triple *annotation* in provenance communities (Green et al., 2007).

that make them prone to overfitting.

To encourage generalization capability, KG embeddings should be able to model inference patterns, i.e., specifications of logical properties that may exist in KGs, which, if learned, empowers further principled inferences (Abboud et al., 2020). This has been extensively studied for binary relational KG embeddings (Trouillon et al., 2016; Sun et al., 2019) but ignored for hyper-relational KGs in which not only primal triples but also qualifiers matter. One of the most important properties is qualifier monotonicity. Given a query, the answer set shrinks or at least does not expand as more qualifiers are added to the query expression. For example, a query $(Einstein, educated_at, ?x)$ with a variable $?x$ corresponds to two answers $\{University\ of\ Zurich, ETH\ Zurich\}$, but a query $((Einstein, educated_at, ?x), \{(degree : B.Sc.)\})$ extended by a qualifier for *degree* will only respond with $\{ETH\ Zurich\}$. Besides, different qualifiers might form logical relationships that the model must respect during inference including qualifier implication (e.g., adding a qualifier that is implicitly implied in the existing qualifiers does not change the truth of a fact) and qualifier mutual exclusion (e.g., adding any two mutually exclusive qualifiers to a fact leads to a contradiction).

In light of this, we propose ShrinkE, a hyper-relational embedding model that allows for modeling these inference patterns. ShrinkE embeds each entity as a point and models a primal triple as a spatio-functional transformation from the head entity to a relation-specific box that entails the possible tails. Each qualifier is modeled as a shrinking of the primal box to a qualifier box. The shrinking of boxes simulates the “monotonicity” of hyper-relational qualifiers, i.e., attaching qualifiers to a primal triple may only narrow down but never enlarges the answer set. The plausibility of a given fact is measured by a point-to-box function that judges whether the tail entity is inside the intersection of all qualifier boxes. Moreover, since each qualifier is associated with a box, the spatial relationships between the qualifier boxes allow for modeling core inference patterns such as qualifier implication and mutual exclusion. We theoretically show the capability of ShrinkE on modeling various inference patterns including (fact-level) monotonicity, triple-level, and qualifier-level inference patterns. Empirically, ShrinkE achieves competitive performance on three benchmarks.

2 Related Work

Related works on hyper-relational KG embeddings can be categorized by their representations of facts. Prominent representations include tuple, key-value pairs, and triple+key-value pairs.

Tuple based Pioneering works view a hyper-relational fact as an n -tuple, a.k.a. n -ary fact, consisting of a single abstract relation r and its n values, i.e., $r(v_1, v_2, \dots, v_n)$. Functional models represent the tuple-based facts by functional mapping. For example, m-TransH (Wen et al., 2016), a generalization of TransH (Wang et al., 2014) to hyper-relational facts, projects all entities onto a relation-specific hyperplane and measures the plausibility as the weighted sum of projected embeddings. RAE (Zhang et al., 2018) improves m-TransH by further modeling the *relatedness* of values. Multi-linear models generalize bilinear models to hyper-relational facts via multi-linear products. For example, Hsimple (Fatemi et al., 2020), m-CP (Fatemi et al., 2020), and GETD (Liu et al., 2020) generalize Simple (Kazemi and Poole, 2018), Canonical Polyadic (CP) decomposition (Trouillon et al., 2017), and TuckER (Balazevic et al., 2019), respectively. GETD only applies to KGs with single-arity relations (Liu et al., 2021b) and S2S (Di et al., 2021) extends it to support mixed arity facts. HypE (Fatemi et al., 2020) encodes hyper-relational facts by positional convolutional filters and evaluates the facts’ plausibility using the multilinear product. However, these models ignore the semantics of relations and loosely represent a combination of all relations of the original fact (Galkin et al., 2020).

Key-value pairs NaLP (Guan et al., 2019) view each hyper-relational fact as a set of key-value pairs, i.e., $\{(k_i : v_i)\}_{i=1}^m$. Convolutional networks are employed to encode the key-value pairs, followed by a multi-layer perceptron (MLP) that measures the compatibility between the key and its values. RAM (Liu et al., 2021b) further models the relatedness between different keys and the relatedness between a key and all involved values. NaLP+ (Guan et al., 2021) improves NaLP by considering type information. However, the key-value-based modeling treats all key-value pairs equally and does not distinguish primal triples from qualifiers.

Triple+key-value pairs NeuInfer (Guan et al., 2020) and HINGE (Rosso et al., 2020) represent a hyper-relational fact as a primary triple combined

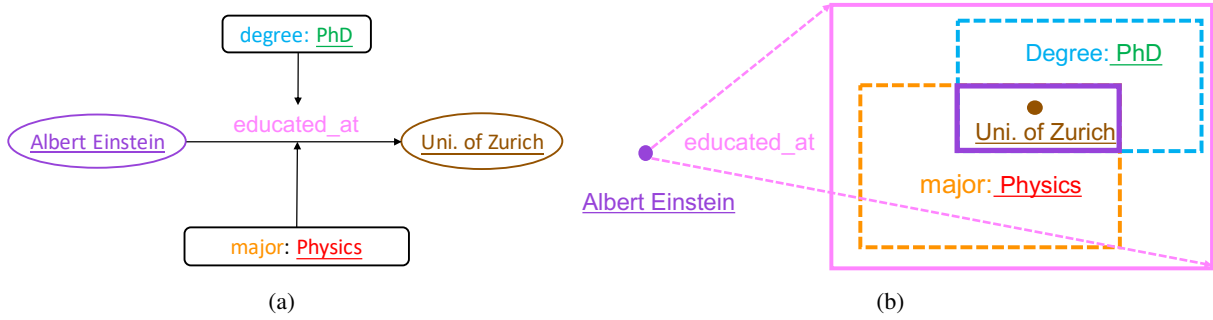


Figure 1: An illustration of the proposed idea. (a) A hyper-relational fact is composed of a primal triple and two key-value qualifiers, in which entities (values) are underlined while relations (keys) are not. (b) An illustration of the proposed hyper-relational KG embedding model ShrinkE. ShrinkE models the primal triple as a relation-specific transformation from the head entity to a query box (*purple*) that entails the possible answer entities. Each qualifier is modeled as a shrinking of the query box (*orange* and *cyan*) such that the shrinking box is a subset of the query box. The shrinking of the box can be viewed as a geometric interpretation of the monotonicity assumption that we follow. The final answer entities are supposed to be in the intersection box of all shrinking boxes.

with a set of the key-value form of qualifiers, i.e., $((h, r, t), \{(k_i : v_i)\}_{i=1}^m)$, which is compatible with the RDF-star standard (Delva et al., 2021) used in modern KGs. Both methods adopt neural networks to obtain the fact validity by measuring the validity of the primary triple and its compatibility with each qualifier. NeuInfer applies MLP while HINGE uses a convolutional network as an encoder. StarE (Galkin et al., 2020) leverages a message passing network, CompGCN (Vashishth et al., 2020), as an encoder to obtain the relation and entity embeddings, which are then fed into a transformer decoder to obtain the validity of facts. Hy-Transformer (Yu and Yang, 2021), GRAN (Wang et al., 2021) and QUAD (Shomer et al., 2022) further improve it with alternative designs of encoders and via auxiliary training tasks. Relatively, these models, though useful, require a large number of parameters and are prone to overfitting.

3 Preliminaries

We view a hyper-relational fact in the form of a primal triple coupled with a set of qualifiers.

Definition 1 (Hyper-relational fact). *Let \mathcal{E} and \mathcal{R} denote the sets of entities and relations, respectively. A hyper-relational fact \mathcal{F} is a tuple $(\mathcal{T}, \mathcal{Q})$, where $\mathcal{T} = (h, r, t)$, $h, t \in \mathcal{E}, r \in \mathcal{R}$ is a primal triple and $\mathcal{Q} = \{(k_i : v_i)\}_{i=1}^m$, $k_i \in \mathcal{R}, v_i \in \mathcal{E}$ is a set of qualifiers. We call the number of involved entities in \mathcal{F} , i.e., $(m + 2)$, the arity of the fact.*

A hyper-relational fact reduces to a triple/binary fact when $m = 0$. When $m > 0$, each qualifier can be viewed as an auxiliary description that contextualizes or specializes the semantics of the primal

triple. In typical open-world settings, facts with the same primal triple might have different numbers of qualifiers. To characterize this property, we introduce the concepts of partial fact and qualifier monotonicity in hyper-relational KGs.

Definition 2 (Partial fact (Guan et al., 2020)). *Given two facts $\mathcal{F}_1 = (\mathcal{T}, \mathcal{Q}_1)$ and $\mathcal{F}_2 = (\mathcal{T}, \mathcal{Q}_2)$ that share the same primal triple. We call \mathcal{F}_1 a partial fact of \mathcal{F}_2 iff $\mathcal{Q}_1 \subseteq \mathcal{Q}_2$.*

In this paper, we follow the monotonicity assumption by restricting the model to respect the monotonicity property.² For this purpose, we consider the monotonicity of query and inference.

Definition 3 (Qualifier monotonicity). *Let $\text{QA}(\cdot)$ denote a query answering model taking a query and a KG as input and outputting the set of answer entities. Given any pair of queries $q_1 = ((h, r, x?), \mathcal{Q}_1)$ and $q_2 = ((h, r, x?), \mathcal{Q}_2)$ that share the same primal triple and $\mathcal{Q}_1 \subseteq \mathcal{Q}_2$, qualifier monotonicity is given iff,*

$$\text{QA}(q_2; \text{KG}) \subseteq \text{QA}(q_1; \text{KG}). \quad (1)$$

Qualifier monotonicity implies that attaching any qualifiers to a query does not enlarge the answer set of the possible tail entities, and inversely, removing the qualifiers from a query can only return more possible tail entities. This implies that if a fact is true, then all its partial facts must also be true (a.k.a. weakening of inference rule), i.e.,

$$(\mathcal{T}, \mathcal{Q}_1) \wedge (\mathcal{Q}_2 \subseteq \mathcal{Q}_1) \rightarrow (\mathcal{T}, \mathcal{Q}_2). \quad (2)$$

²Some kinds of qualifiers may represent semantically opaque contexts. For instance, $((\text{Crimea}, \text{belongs_to}, \text{Russia}), \{(\text{said_by}, \text{Putin})\})$ does not imply the primary triple and should therefore be excluded.

4 Shrinking Embeddings for Hyper-Relational KGs

We aim to design a scoring function $f(\cdot)$ taking the embeddings of facts as input so that the output values respect desired logical properties. To this end, we introduce primal triple embedding and qualifier embedding, respectively.

4.1 Primal Triple Embedding

We represent each entity as a point $\mathbf{e} \in \mathbb{R}^d$. Each primal relation r is modeled as a spatio-functional transformation $\mathcal{B}_r : \mathbb{R}^d \rightarrow \text{Box}(d)$ that maps the head $\mathbf{e}_h \in \mathbb{R}^d$ to a d -dimensional box in $\text{Box}(d)$ with $\text{Box}(d)$ being the set of boxes in \mathbb{R}^d . Each box can be parameterized by a lower left point $\mathbf{m} \in \mathbb{R}^d$ and an upper right point $\mathbf{M} \in \mathbb{R}^d$, given by

$$\text{Box}^d(\mathbf{m}, \mathbf{M}) = \{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{m}_i \leq \mathbf{x}_i \leq \mathbf{M}_i, i = 1, \dots, d\}. \quad (3)$$

We leave the superscript of Box^d away if it is clear from context. and call the transformed box a query box. Intuitively, all points in the query box correspond to the possible answer tail entities. Hence, the query box can be viewed as a geometric embedding of the answer set. Note that a query could result in an empty answer set. In order to capture such property, we do not exclude empty boxes that correspond to queries with empty answer set. Empty boxes are covered by the cases where there exists a dimension i such that $\mathbf{m}_i \geq \mathbf{M}_i$.

Point-to-box transform The spatio-functional point-to-box transformation \mathcal{B} is composed of a relation-specific point transformation $\mathcal{H}_r : \mathbb{R}^d \rightarrow \mathbb{R}^d$ that transforms the head point \mathbf{e}_h to a new point, and a relation-specific spanning that spans the transformed point to a box, formally given by

$$\mathcal{B}_r(\mathbf{e}_h) = \text{Box}(\mathcal{H}_r(\mathbf{e}_h) - \tau(\delta_r), \mathcal{H}_r(\mathbf{e}_h) + \tau(\delta_r)), \quad (4)$$

where $\delta_r \in \mathbb{R}^n$ is a relation-specific spanning/offset vector, and $\tau_t(\mathbf{x}) = t \log(1 + e^{\mathbf{x}/t})$ with t being a temperature hyperparameter, is a *softplus* function that enforces the spanned box to be non-empty.

The point transformation function \mathcal{H}_r could be any functions that are used in other KG embedding models such as translation used in TransE (Bordes et al., 2013) and rotations used in RotatE (Sun et al., 2019). Hence, our model is highly flexible and effective at embedding primal triples. To allow for

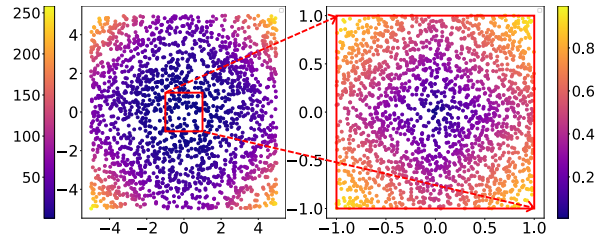


Figure 2: An illustration of the point-to-box distance. The distance (visualized by color maps) grows slowly when the point is inside of the box (right) while growing faster when the point is outside of the box (left).

capturing multiple triple-level inference patterns such as symmetry, inversion, and composition, we combine translation and rotation, and formulate \mathcal{H}_r as

$$\mathcal{H}_r(\mathbf{e}_h) = \Theta_r \mathbf{e}_h + \mathbf{b}_r \quad (5)$$

where Θ_r is a rotation matrix and \mathbf{b}_r is a translation vector. We parameterize the rotation matrix by a block diagonal matrix $\Theta_r = \text{diag}(\mathbf{G}(\theta_{r,1}), \dots, \mathbf{G}(\theta_{r, \frac{d}{2}}))$, where

$$\mathbf{G}(\theta) = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}. \quad (6)$$

Point-to-box distance The validity of a primal triple (h, r, t) is then measured by judging whether the tail entity point \mathbf{e}_t is geometrically inside of the query box. Given a query box $\text{Box}^n(\mathbf{m}, \mathbf{M})$ and an entity point $\mathbf{e} \in \mathbb{R}^d$, we denote the center point as $\mathbf{c} = \frac{\mathbf{m} + \mathbf{M}}{2}$. Let $|\cdot|$ denote the L1 norm and $\max(\cdot)$ denote an element-wise maximum operation. The point-to-box distance is given by

$$D(\mathbf{e}, \text{Box}(\mathbf{m}, \mathbf{M})) = \frac{|\mathbf{e} - \mathbf{c}|_1}{|\max(\mathbf{0}, \mathbf{M} - \mathbf{m})|_1} + (|\mathbf{e} - \mathbf{m}|_1 + |\mathbf{e} - \mathbf{M}|_1 - |\max(\mathbf{0}, \mathbf{M} - \mathbf{m})|_1)^2. \quad (7)$$

Fig. 2 visualizes the distance function. Intuitively, in cases where the point is in the query box, the distance grows relatively slowly and inversely correlates with the box size. In cases where the point is outside the box, the distance grows fast.

4.2 Qualifier Embedding

Conceptually, qualifiers add information to given primary facts potentially allowing for additional inferences, but never for the retraction of inferences, reflecting the monotonicity of the representational paradigm. Corresponding to the non-declining number of inferences, the number of possible models for this representation shrinks, which can be intuitively reflected by a reduced size of boxes incurred by adding qualifiers.

Box Shrinking To geometrically mimic this property in the embedding space, we model each qualifier ($k : v$) as a "shrinking" of the query box. Given a box $\text{Box}(\mathbf{m}, \mathbf{M})$, a shrinking is defined as a box-to-box transformation $\mathcal{S} : \text{Box} \rightarrow \text{Box}$ that potentially shrinks the volume of the box while not moving the resulting box outside of the source box. Let $\mathbf{L} = (\mathbf{M} - \mathbf{m})$ denote the side length vector, box shrinking is defined by

$$\begin{aligned} \mathcal{S}_{r,k,v}(\text{Box}(\mathbf{m}, \mathbf{M})) = \\ \text{Box}(\mathbf{m} + \sigma(\mathbf{s}_{r,k,v}) \odot \mathbf{L}, \mathbf{M} - \sigma(\mathbf{S}_{r,k,v}) \odot \mathbf{L}), \end{aligned} \quad (8)$$

where $\mathbf{s}_{r,k,v} \in \mathbb{R}^n$ and $\mathbf{S}_{r,k,v} \in \mathbb{R}^n$ are the "shrinking" vectors for the lower left corner and the upper right corner, respectively. σ is a *sigmoid* function and \odot is element-wise vector multiplication. The resulting box, including the case of empty box, is always inside the query box, i.e., $\mathcal{S}(\text{Box}^n(\mathbf{m}, \mathbf{M})) \subseteq \text{Box}^n(\mathbf{m}, \mathbf{M})$, which exactly resembles the qualifier monotonicity.

We use r, k, v as the indices of the shrinking vectors because the shrinking of the box should depend on the relatedness between the primal relation and the qualifier. For example, if a qualifier (*degree : bachelor*) is highly related to the primal relation *educated_at*, the scale of the shrinking vectors should be small as it adds a weak constraint to the triple. If the qualifier is unrelated to the primal relation, e.g., (*degree : bachelor*) and *born_in*, the shrinking might even enforce an empty box.

To learn the shrinking vectors, we leverage an MLP layer that takes the primal relation and key-value qualifier as input and outputs the shrinking vectors defined by $\mathbf{s}_{r,k,v}, \mathbf{S}_{r,k,v} = \text{MLP}(\text{concat}(r_\theta, k_\theta, v_\theta))$ where $r_\theta, k_\theta, v_\theta$ are the embeddings of r, k, v , respectively.

4.3 Scoring function and learning.

Scoring function The score of a given hyper-relational fact is defined by

$$f(((h, r, t), \mathcal{Q})) = D(\mathbf{e}_t, \text{Box}_{\mathcal{Q}}(\mathbf{m}, \mathbf{M})), \quad (9)$$

where $\text{Box}_{\mathcal{Q}}(\mathbf{m}, \mathbf{M})$ denotes the target box that is calculated by the intersection of all shrinking boxes of the qualifier set \mathcal{Q} . The intersection of n boxes can be calculated by taking the maximum of lower left points of all boxes and taking the minimum of upper right points of all boxes, given by

$$\mathcal{I}(\text{Box}_1, \dots, \text{Box}_n) = \text{Box}\left(\max_{i \in 1, \dots, n} \mathbf{m}_i, \min_{i \in 1, \dots, n} \mathbf{M}_i\right). \quad (10)$$

Note that if there is no intersection between boxes, this intersection operation still works as it results in an empty box. The intersection of boxes is a permutation-invariant operation, implying that perturbing the order of qualifiers does not change the plausibility of the facts.

Learning As a standard data augmentation strategy, we add reciprocal relations (t', r^{-1}, h') for the primary triple in each hyper-relational fact. For each positive fact in the training set, we generate n_{neg} negative samples by corrupting a subject/tail entity with randomly selected entities from \mathcal{E} . We adopt the cross-entropy loss to optimize the model via the Adam optimizer, which is given by

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^N \left(y_i \log(p_i) + \sum_{i=1}^{n_{\text{neg}}} (1 - y_i) \log(1 - p_i) \right), \quad (11)$$

where N denotes the total number of facts in the training set. y_i is a binary indicator denoting whether a fact is true or not. $p_i = \sigma(f(\mathcal{F}))$ is the predicted score of a fact \mathcal{F} with σ being the *sigmoid* function.

5 Theoretical Analysis

Analyzing and modeling inference patterns is of great importance for KG embeddings because it enables generalization capability, i.e., once the patterns are learned, new facts that respect the patterns can be inferred. An inference pattern is a specification of a logical property that may exist in a KG. Formally, an inference pattern is a logical form $\psi \rightarrow \phi$ with ψ and ϕ being the body and head, implying that if the body is satisfied then the head must also be satisfied.

In this section, we analyze the theoretical capacity of ShrinkE for modeling inference patterns. All proofs of propositions are in Appendix B.

Fact-level inference pattern (monotonicity)

The following proposition shows that ShrinkE is able to model monotonicity.

Proposition 1. *Given any two facts $\mathcal{F}_1 = (\mathcal{T}, \mathcal{Q}_1)$ and $\mathcal{F}_2 = (\mathcal{T}, \mathcal{Q}_2)$ where $\mathcal{Q}_2 \subseteq \mathcal{Q}_1$, i.e., \mathcal{F}_2 is a partial fact of \mathcal{F}_1 , the output of the scoring function $f(\cdot)$ of ShrinkE satisfy the constraint $f(\mathcal{F}_2) \geq f(\mathcal{F}_1)$.*

Triple-level inference patterns Prominent triple-level inference patterns include symmetry $(h, r, t) \rightarrow (t, r, h)$, anti-symmetry $(h, r, t) \rightarrow \neg(h, r, t)$, inver-

	All facts	Higher-arity facts (%)	Entities	Relations	Train	Dev	Test
JF17K	100,947	46,320 (45.9%)	28,645	501	76,379	–	24,568
WikiPeople	382,229	44,315 (11.6%)	47,765	193	305,725	38,223	38,281
WD50k	236,507	32,167 (13.6%)	47,156	532	166,435	23,913	46,159
WD50K(33)	102,107	31,866 (31.2%)	38,124	475	73,406	10,568	18,133
WD50K(66)	49,167	31,696 (64.5%)	27,347	494	35,968	5,154	8,045
WD50K(100)	31,314	31,314 (100%)	18,792	279	22,738	3,279	5,297

Table 1: Dataset statistics, where the columns indicate the number of all facts, hyper-relational facts with the number of qualifiers $m > 0$, entities, relations, and facts in train/dev/test sets, respectively.

sion $(h, r_1, t) \rightarrow (t, r_2, h)$, composition $(e_1, r_1, e_2) \wedge (e_2, r_2, e_3) \rightarrow (e_1, r_3, e_3)$, relation implication $(h, r_1, t) \rightarrow (h, r_2, t)$, relation intersection $(h, r_1, t) \wedge (h, r_2, t) \rightarrow (h, r_3, t)$, and relation mutual exclusion $(h, r_1, t) \wedge (h, r_2, t) \rightarrow \perp$. All these triple-level inference patterns also exist in hyper-relational facts when their qualifiers are the same, e.g., hyper-relational symmetry means $((h, r, t), \mathcal{Q}) \rightarrow ((t, r, h), \mathcal{Q})$. Proposition 2 states that ShrinkE is able to infer all of them.

Proposition 2. *ShrinkE is able to infer hyper-relational symmetry, anti-symmetry, inversion, composition, relation implication, relation intersection, and relation exclusion.*

Qualifier-level inference pattern In hyper-relational KGs, inference patterns not only exist at the triple level but also at the level of qualifiers.

Definition 4 (qualifier implication). *Given two qualifiers q_i and q_j , q_i is said to imply q_j , i.e., $q_i \rightarrow q_j$ iff for any fact $\mathcal{F} = (\mathcal{T}, \mathcal{Q})$, if attaching q_i to \mathcal{Q} results in a true (resp. false) fact, then attaching q_j to $\mathcal{Q} \cup \{q_i\}$ also results in a true (resp. false) fact. Formally, $q_i \rightarrow q_j$ implies*

$$\forall \mathcal{T}, \mathcal{Q} : (\mathcal{T}, \mathcal{Q} \cup \{q_i\}) \rightarrow (\mathcal{T}, \mathcal{Q} \cup \{q_i, q_j\}). \quad (12)$$

Definition 5 (qualifier exclusion). *Two qualifiers q_i, q_j are said to be mutually exclusive iff for any fact $\mathcal{F} = (\mathcal{T}, \mathcal{Q})$, by attaching q_i, q_j to the qualifier set of \mathcal{F} , the new fact $\mathcal{F}' = (\mathcal{T}, \mathcal{Q} \cup \{q_i, q_j\})$ is false, meaning that they lead to a contradiction, i.e., $q_i \wedge q_j \rightarrow \perp$. Formally, $q_i \wedge q_j \rightarrow \perp$ implies*

$$\forall \mathcal{T}, \mathcal{Q} : (\mathcal{T}, \mathcal{Q} \cup \{q_i, q_j\}) \rightarrow \perp \quad (13)$$

Note that if two qualifiers q_i, q_j are neither mutually exclusive nor forming implication pair, then q_i, q_j are said to be overlapping, a state between implication and mutual exclusion. Qualifier overlapping, in our case, can be captured by box intersection/overlapping. Qualifier overlapping itself

does not form any logical property in the form of $\psi \rightarrow \phi$. However, when involving three qualifiers and two of them overlap, qualifier intersection can be modeled.

Definition 6 (qualifier intersection). *A qualifier q_k is said to be an intersection of two qualifiers q_i, q_j iff for any fact $\mathcal{F} = (\mathcal{T}, \mathcal{Q})$, if attaching q_i, q_j to \mathcal{Q} results in a true (resp. false) fact, then by replacing $\{q_i, q_j\}$ with q_k , the truth value of the fact does not change. Namely, $q_i \wedge q_j \rightarrow q_k$ implies*

$$\forall \mathcal{T}, \mathcal{Q} : (\mathcal{T}, \mathcal{Q} \cup \{q_i, q_j\}) \rightarrow (\mathcal{T}, \mathcal{Q} \cup \{q_k\}). \quad (14)$$

Apparently, qualifier intersection $q_i \wedge q_j \rightarrow q_k$ necessarily implies qualifier implications $q_i \rightarrow q_k$ and $q_j \rightarrow q_k$. Hence, qualifier intersection can be viewed as a combination of two qualifier implications, and this can be generalized to $q_1 \wedge q_2 \wedge \dots \rightarrow q_k$. Proposition 3 shows that ShrinkE is able to infer qualifier implication, exclusion, and composition.

Proposition 3. *ShrinkE is able to infer qualifier implication, mutual exclusion, and intersection.*

6 Evaluation

In this section, we evaluate the effectiveness of ShrinkE on hyper-relational link prediction tasks.

6.1 Experimental Setup

Datasets. We conduct link prediction experiment on three hyper-relational KGs: JF17K (Wen et al., 2016), WikiPeople (Guan et al., 2019), and WD50k (Galkin et al., 2020). JF17K is extracted from Freebase while WikiPeople and WD50k are extracted from Wikidata. In WikiPeople and WD50k, only 11.6% and 13.6% of the facts, respectively, contain qualifiers, while the remaining facts contain only triples (after dropping statements containing literals in WikiPeople, only 2.6% facts contain qualifiers). For better comparison, we also consider

Method	WikiPeople (2.6)			JF17K (45.9)			WD50K (13.6)		
	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10
m-TransH	0.063	0.063	0.300	0.206	0.206	0.463	—	—	—
RAE	0.059	0.059	0.306	0.215	0.215	0.469	—	—	—
NaLP-Fix	0.420	0.343	0.556	0.245	0.185	0.358	0.177	0.131	0.264
NeuInfer	0.350	0.282	0.467	0.451	0.373	0.604	—	—	—
HINGE	0.476	0.415	0.585	0.449	0.361	0.624	0.243	0.176	0.377
Transformer	0.469	0.403	0.586	0.512	0.434	0.665	0.264	0.194	0.401
BoxE	0.395	0.293	0.503	0.560	0.472	0.722	—	—	—
StarE	0.491	0.398	0.648	0.574	0.496	0.725	0.349	<u>0.271</u>	0.496
ShrinkE	<u>0.485</u>	0.431	<u>0.601</u>	0.589	0.506	0.749	<u>0.345</u>	0.275	<u>0.482</u>

Table 2: Link prediction results on three benchmarks with the number in the parentheses denoting the ratio of facts with qualifiers. Baseline results are taken from Galkin et al. (2020).

Method	WD50K (33)			WD50K (66)			WD50K (100)		
	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10
NaLP-Fix	0.204	0.164	0.277	0.334	0.284	0.423	0.458	0.398	0.563
HINGE	0.253	0.190	0.372	0.378	0.307	0.512	0.492	0.417	0.636
Transformer	0.276	0.227	0.371	0.404	0.352	0.502	0.562	0.499	0.677
StarE	<u>0.331</u>	<u>0.268</u>	0.451	<u>0.481</u>	<u>0.420</u>	<u>0.594</u>	0.654	<u>0.588</u>	<u>0.777</u>
ShrinkE	0.336	0.272	<u>0.449</u>	0.511	0.422	0.611	0.695	0.629	0.814

Table 3: Link prediction results on WD50K splits with the number in the parentheses denoting the ratio of facts with qualifiers. Baseline results are taken from Galkin et al. (2020).

three splits of WD50K that contain a higher percentage of triples with qualifiers. The three splits are WD50K(33), WD50K(66), and WD50K(100), which contain 33%, 66%, and 100% facts with qualifiers, respectively. Statistics of the datasets are given in Table 1. We conjecture that the performance on WikiPeople and WD50k will be dominated by the scores of triple-only facts while the performance on the variants of WD50k will be dominated by the modeling of qualifiers. We conjecture that WD50K will be a more challenging benchmark than JF17K and WikiPeople. Besides, WD50K still contains only a small percentage (13.6%) of facts that contain qualifiers. Since JF17K does not provide a validation set, we split 20% of facts from the training set as the validation set. Details of the three datasets are given in Table 1.

Environments and hyperparameters We implement ShrinkE with Python 3.9 and Pytorch 1.11, and train our model on one Nvidia A100 GPU with 40GB of VRAM. We use Adam optimizer with a batch size of 128 and an initial learning rate of 0.0001. For negative sampling, we follow the strategy used in StarE (Galkin et al., 2020) by randomly corrupting the head or tail entity in the primal triple.

Different from HINGE (Rosso et al., 2020) and NeuInfer (Guan et al., 2020) that score all potential facts one by one that takes an extremely long time for evaluation, ShrinkE ranks each target answer against all candidates in a single pass and significantly reduces the evaluation time. We search the dimensionality from [50, 100, 200, 300] and the best one is 200. We set the temperature parameter to be $t = 1.0$. We use the label smoothing strategy and set the smoothing rate to be 0.1. We repeat all experiments for 5 times with different random seeds and report the average values, the error bars are relatively small and are omitted. Code is available at ³.

Baselines We compare ShrinkE against various models, including m-TransH (Wen et al., 2016), RAE (Zhang et al., 2018), NaLP-Fix (Rosso et al., 2020), HINGE (Rosso et al., 2020), NeuInfer (Guan et al., 2020), BoxE (Abboud et al., 2020), Transformer and StarE (Galkin et al., 2020). Note that we exclude Hy-Transformer (Yu and Yang, 2021), GRAN (Wang et al., 2021) and QUAD (Shomer et al., 2022) for comparison because 1) they are heavily based on StarE and Transformer;

³<https://github.com/xiongbo010/ShrinkE>

Method	MRR	H@ 1	H@ 10
ShrinkE (w/o translation)	0.583	0.495	0.729
ShrinkE (w/o rotation)	0.581	0.497	0.724
ShrinkE (w/o shrinking)	0.571	0.490	0.711
ShrinkE	0.589	0.506	0.749

Table 4: The performance of ShrinkE by removing one relational component on JF17K.

and 2) they leverage auxiliary training tasks, which can also be incorporated into our framework and we leave as one future work.

Evaluation We strictly follow the settings of Galkin et al. (2020), where the aim is to predict a missing head/tail entity in a hyper-relational fact. We consider the widely used ranking-based metrics for link prediction: mean reciprocal rank (MRR) and H@K (K=1,10). For ranking calculation, we consider the filtered setting by filtering the facts existing in the training and validation sets (Bordes et al., 2013).

6.2 Main Results and Analysis

Table 2 and Table 3 summarize the performances of all approaches on the six datasets. Overall, ShrinkE achieves either the best or the second-best results against all baselines, showcasing the expressivity and capability of ShrinkE on hyper-relational link prediction. In particular, We observe that ShrinkE outperforms all baselines on JF17K and the three variants of WD50K with a high ratio of facts containing qualifiers while achieving highly competitive results on WikiPeople and the original version of WD50K that contain fewer facts with qualifiers. Interestingly, we find that the performance gains increase when increasing the ratio of facts containing qualifiers. On WD50K (100) where 100% facts contain qualifiers, the performance gain of ShrinkE is most significant across all metrics (6.2%, 6.9%, and 4.7% improvements over MRR, H@1, and H@10, respectively). We believe this is because that ShrinkE is excellent at modeling qualifiers due to its explicit modeling of inference patterns.

Case analysis Table 5 shows some examples of qualifier implication pairs recovered by our learned embeddings. Note that exclusions pairs are ubiquitous (i.e., most of the random qualifiers are mutually exclusive) and hence we do not analyze them. We find that some qualifier implications happen when they are about geographic information and involve geographic inclusion such as *Monte Carlo*

body	head
(residence: Monte Carlo)	(country, Monaco)
(residence: Belgrade)	(country, Serbia)
(owned_by: X)	(of, voting interest)
(emergency phone number: Y)	(has_use, police)
(emergency phone number: Z)	(has_use, fire department)
(used_by: software)	(via, operating_system)

Table 5: Example pairs of qualifiers with implication relations (body \rightarrow head). $X \in [Eric\ Schmidt, Mark\ Zuckerberg, Dustin\ Moskovitz, Larry\ Page]$ denotes a CEO name of a company. $Y \in [112, 115, 113, \dots]$ and $Z \in [912, 18, 192, \dots]$ are emergency numbers involving *police* and *fire department*, respectively. Qualifier exclusion pairs are ubiquitous and are hence omitted.

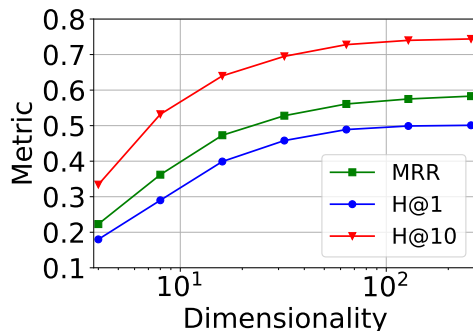


Figure 3: Performance of ShrinkE with different dimensions $d = [4, 8, 16, 32, 64, 128, 256]$ on JF17K.

is in *Monaco*. Interestingly, we find that qualifiers associated with key *owned_by* imply (*of, voting interest*), and qualifiers with key *emergency phone number* imply (*has_use, police*) or (*has_use, fire department*), which conceptually make sense.

6.3 Ablations and Parameter Sensitivity

Impact of relational components To determine the importance of each component in relational modeling, we conduct an ablation study by considering three versions of ShrinkE in which one of the components (translation, rotation, and shrinking) is removed. Table 4 shows that the removal of each component of the relational transformation leads to a degradation in performance, validating the importance of each component. In particular, by removing the qualifier shrinking, which is the main contribution of our framework, the performance reduces 3% and 5% in MRR and H@10, respectively, showcasing the usefulness of modeling qualifiers as shrinking. The removals of translation and rotation both result in around 1% and 2% reduction in MRR and H@10, respectively.

Impact of dimensionality We conduct experiments on JF17K under a varied number of dimen-

sions $d = [4, 8, 16, 32, 64, 128, 256]$. As Fig. 3 depicts, the performance increases when increasing the number of dimensions. However, the growth trend gradually flattens with the increase of dimensions and it achieves comparable performance when the dimension is higher than 128.

6.4 Discussion

Comparison with neural network models

Heavy neural network models such as GRAN (Wang et al., 2021) and QUAD (Shomer et al., 2022) are built on relational GNNs and/or Transformers and require a large number of parameters. In contrast, ShrinkE is a neuro-symbolic model that requires only one MLP layer and a much smaller number of parameters. The logical modelling of ShrinkE makes it more explainable than GNN-based and Transformer-based methods.

Comparison with other box embeddings in KGs

ShrinkE is the first to not only represent hyper-relational facts, but also explicitly model the logical properties of these facts. ShrinkE is different from previous box embedding methods (Abboud et al., 2020) of KGs in three key modules: 1) our point-to-box transform function modelling triple inference patterns; 2) a new point-to-box distance function; and 3) we introduce box shrinking to model qualifier-level inference patterns. Moreover, we provide a comprehensive theoretical analysis of ShrinkE on modelling various logical properties.

7 Conclusion

We present a novel hyper-relational KG embedding model ShrinkE. ShrinkE models a primal triple as a spatio-functional transformation while modeling each qualifier as a shrinking that monotonically narrows down the answer set. We proved that ShrinkE is able to spatially infer core inference patterns at different levels including triple-level, fact-level, and qualifier-level. Experimental results on three benchmarks demonstrate the advantages of ShrinkE in predicting hyper-relational links.

Limitations

Currently, the main goal of ShrinkE is to model inference patterns directly in the embedding space for hyper-relational KGs and we do not explore more advanced training strategies that have recently been proposed. For example, recent works (Yu and Yang, 2021; Wang et al., 2021; Shomer et al., 2022) have demonstrated that adding auxiliary training tasks,

e.g., the task of predicting qualifier entities, can further improve the overall performance. We believe such auxiliary training tasks can also benefit ShrinkE and we leave it as future work. Another limitation of ShrinkE, though rarely happens, is that when dealing with semantically opaque contexts, the monotonicity assumption might not hold. In that case, we need ad-hoc solutions. One simple way is to explicitly distinguish semantically transparent and semantically opaque contexts.

Ethics Statement

The authors declare that they have no conflicts of interest. This article does not contain any studies involving business data and personal information. Our experimentation does not involve any ethical concerns. However, similar to other models, when deploying our link prediction model to real-world applications such as online recommendation systems, the prediction might be biased or unfair to some ethnic/gender groups. We advise researchers in the community to look into bias (Bourli and Pitoura, 2020) and fairness (Fu et al., 2020) in KGs.

Acknowledgement

The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Bo Xiong. Bo Xiong is funded by the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No: 860801. Mojtaba Nayyeri is funded by the German Federal Ministry for Economic Affairs and Climate Action under Grant Agreement Number 01MK20008F (Service-Meister). This research was partially funded by the Ministry of Science, Research, and the Arts (MWK) Baden-Württemberg, Germany, within the Artificial Intelligence Software Academy (AISA) and the German Research Foundation (DFG) via grant agreement number STA 572/18-1 (Open Argument Mining). We acknowledge the support by the Stuttgart Center for Simulation Science (SimTech).

References

- Ralph Abboud, İsmail İlkan Ceylan, Thomas Lukasiewicz, and Tommaso Salvatori. 2020. Boxe: A box embedding model for knowledge base completion. In *NeurIPS*.
- Dörthe Arndt, Jeen Broekstra, Bob DuCharme, Ora Lassila, Peter F. Patel-Schneider, Eric

- Prud'hommeaux, Jr. Ted Thibodeau, and Bryan Thompson. 2021. Rdf-star and sparql-star. In *Final Community Group Report*.
- Ivana Balazevic, Carl Allen, and Timothy M. Hospedales. 2019. Tucker: Tensor factorization for knowledge graph completion. In *EMNLP/IJCNLP (1)*, pages 5184–5193. Association for Computational Linguistics.
- Kurt Bollacker, Robert Cook, and Patrick Tufts. 2007. Freebase: A shared database of structured general human knowledge. In *AAAI*, volume 7, pages 1962–1963.
- Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *NIPS*, pages 2787–2795.
- Styliani Bourli and Evaggelia Pitoura. 2020. Bias in knowledge graph embeddings. In *2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 6–10. IEEE.
- Ines Chami, Adva Wolf, Da-Cheng Juan, Frederic Sala, Sujith Ravi, and Christopher Ré. 2020. Low-dimensional hyperbolic knowledge graph embeddings. In *ACL*, pages 6901–6914. Association for Computational Linguistics.
- Yankai Chen, Menglin Yang, Yingxue Zhang, Mengchen Zhao, Ziqiao Meng, Jianye Hao, and Irwin King. 2022. Modeling scale-free graphs with hyperbolic geometry for knowledge-aware recommendation. In *WSDM*, pages 94–102. ACM.
- Thomas Delva, Julián Arenas-Guerrero, Ana Iglesias-Molina, Oscar Corcho, David Chaves-Fraga, and Anastasia Dimou. 2021. Rml-star: A declarative mapping language for rdf-star generation. In *ISWC*, pages 1–5.
- Shimin Di, Quanming Yao, and Lei Chen. 2021. Searching to sparsify tensor decomposition for n-ary relational data. In *WWW*, pages 4043–4054. ACM / IW3C2.
- Bahare Fatemi, Perouz Taslakian, David Vázquez, and David Poole. 2020. Knowledge hypergraphs: Prediction beyond binary relations. In *IJCAI*, pages 2191–2197. ijcai.org.
- Zuohui Fu, Yikun Xian, Ruoyuan Gao, Jieyu Zhao, Qiaoying Huang, Yingqiang Ge, Shuyuan Xu, Shijie Geng, Chirag Shah, Yongfeng Zhang, et al. 2020. Fairness-aware explainable recommendation over knowledge graphs. In *SIGIR*, pages 69–78.
- Mikhail Galkin, Priyansh Trivedi, Gaurav Maheshwari, Ricardo Usbeck, and Jens Lehmann. 2020. Message passing for hyper-relational knowledge graphs. In *EMNLP (1)*, pages 7346–7359. Association for Computational Linguistics.
- Todd J Green, Grigoris Karvounarakis, and Val Tannen. 2007. Provenance semirings. In *Proceedings of the twenty-sixth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 31–40.
- Saiping Guan, Xiaolong Jin, Jiafeng Guo, Yuanzhuo Wang, and Xueqi Cheng. 2021. Link prediction on n-ary relational data based on relatedness evaluation. *TKDE*.
- Saiping Guan, Xiaolong Jin, Jiafeng Guo, Yuanzhuo Wang, and Xueqi Cheng. 2020. Neuinfer: Knowledge inference on n-ary facts. In *ACL*, pages 6141–6151. Association for Computational Linguistics.
- Saiping Guan, Xiaolong Jin, Yuanzhuo Wang, and Xueqi Cheng. 2019. Link prediction on n-ary relational data. In *WWW*, pages 583–593. ACM.
- Yunjie He, Mojtaba Nayyeri, Bo Xiong, Evgeny Kharlamov, and Steffen Staab. 2023. Modeling relational patterns for logical query answering over knowledge graphs. *CoRR*, abs/2303.11858.
- Seyed Mehran Kazemi and David Poole. 2018. Simple embedding for link prediction in knowledge graphs. In *NeurIPS*, pages 4289–4300.
- Maxat Kulmanov, Wang Liu-Wei, Yuan Yan, and Robert Hoehndorf. 2019. EL embeddings: Geometric construction of models for the description logic EL++. In *IJCAI*, pages 6103–6109. ijcai.org.
- Yu Liu, Quanming Yao, and Yong Li. 2020. Generalizing tensor decomposition for n-ary relational knowledge bases. In *WWW*, pages 1104–1114. ACM / IW3C2.
- Yu Liu, Quanming Yao, and Yong Li. 2021a. Role-aware modeling for n-ary relational knowledge bases. In *WWW*, pages 2660–2671. ACM / IW3C2.
- Yu Liu, Quanming Yao, and Yong Li. 2021b. Role-aware modeling for n-ary relational knowledge bases. In *Proceedings of the Web Conference 2021*, pages 2660–2671.
- Jiaying Lu, Jiaming Shen, Bo Xiong, Wenjing Ma, Steffen Staab, and Carl Yang. 2023. Hiprompt: Few-shot biomedical knowledge fusion via hierarchy-oriented prompting. In *SIGIR*. ACM.
- Denis Lukovnikov, Asja Fischer, Jens Lehmann, and Sören Auer. 2017. Neural network-based question answering over knowledge graphs on word and character level. In *WWW*, pages 1211–1220.
- Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A three-way model for collective learning on multi-relational data. In *ICML*, pages 809–816. Omnipress.
- Hongyu Ren, Weihua Hu, and Jure Leskovec. 2020. Query2box: Reasoning over knowledge graphs in vector space using box embeddings. In *ICLR*. OpenReview.net.

- Paolo Rosso, Dingqi Yang, and Philippe Cudré-Mauroux. 2020. Beyond triplets: Hyper-relational knowledge graph embedding for link prediction. In *WWW*, pages 1885–1896. ACM / IW3C2.
- Harry Shomer, Wei Jin, Juan-Hui Li, Yao Ma, and Jiliang Tang. 2022. Learning representations for hyper-relational knowledge graphs. *CoRR*, abs/2208.14322.
- Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. In *ICLR (Poster)*. OpenReview.net.
- Théo Trouillon, Christopher R Dance, Éric Gaussier, Johannes Welbl, Sebastian Riedel, and Guillaume Bouchard. 2017. Knowledge graph completion via complex tensor factorization. *JMLR*, 18:1–38.
- Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *ICML*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 2071–2080. JMLR.org.
- Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha P. Talukdar. 2020. Composition-based multi-relational graph convolutional networks. In *ICLR*. OpenReview.net.
- Luke Vilnis, Xiang Li, Shikhar Murty, and Andrew McCallum. 2018. Probabilistic embedding of knowledge graphs with box lattice measures. In *ACL (1)*, pages 263–272. Association for Computational Linguistics.
- Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.
- Quan Wang, Haifeng Wang, Yajuan Lyu, and Yong Zhu. 2021. Link prediction on n-ary relational facts: A graph-based approach. In *ACL/IJCNLP (Findings)*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 396–407. Association for Computational Linguistics.
- Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *AAAI*, pages 1112–1119. AAAI Press.
- Jianfeng Wen, Jianxin Li, Yongyi Mao, Shini Chen, and Richong Zhang. 2016. On the representation and embedding of knowledge bases beyond binary relations. In *IJCAI*, pages 1300–1307. IJCAI/AAAI Press.
- Bo Xiong, Michael Cochez, Mojtaba Nayeri, and Steffen Staab. 2022a. Hyperbolic embedding inference for structured multi-label prediction. In *NeurIPS*.
- Bo Xiong, Mojtaba Nayeri, Ming Jin, Yunjie He, Michael Cochez, Shirui Pan, and Steffen Staab. 2023. Geometric relational embeddings: A survey. *CoRR*, abs/2304.11949.
- Bo Xiong, Nico Potyka, Trung-Kien Tran, Mojtaba Nayeri, and Steffen Staab. 2022b. Faithful embeddings for el++ knowledge bases. In *ISWC*, volume 13489 of *Lecture Notes in Computer Science*, pages 22–38. Springer.
- Bo Xiong, Nico Potyka, Trung-Kien Tran, Mojtaba Nayeri, and Steffen Staab. 2022c. Faithful embeddings for el++ knowledge bases. *ISWC*, abs/2201.09919.
- Bo Xiong, Shichao Zhu, Mojtaba Nayeri, Chengjin Xu, Shirui Pan, Chuan Zhou, and Steffen Staab. 2022d. Ultrahyperbolic knowledge graph embeddings. In *KDD*, pages 2130–2139. ACM.
- Donghan Yu and Yiming Yang. 2021. Improving hyper-relational knowledge graph completion. *CoRR*, abs/2104.08167.
- Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. 2016. Collaborative knowledge base embedding for recommender systems. In *SIGKDD*, pages 353–362.
- Richong Zhang, Junpeng Li, Jiajie Mei, and Yongyi Mao. 2018. Scalable instance reconstruction in knowledge bases via relatedness affiliated embedding. In *WWW*, pages 1185–1194. ACM.
- Zhanqiu Zhang, Jie Wang, Jiajun Chen, Shuiwang Ji, and Feng Wu. 2021. Cone: Cone embeddings for multi-hop reasoning over knowledge graphs. In *NeurIPS*, pages 19172–19183.

A Supplemental Related Works

We survey some supplemental related work on binary relational KG embeddings and geometric relational embeddings.

Binary relational KG embeddings Most of the existing KG embedding methods consider binary relational KGs where each fact is represented in the form of triple (h, r, t) . Prominent examples include the *additive* (or *translational*) family such as TransE (Bordes et al., 2013) that models each fact as a translation $\mathbf{s} + \mathbf{r} \approx \mathbf{o}$, and the *multiplicative* (or *bilinear*) family such as RESCAL (Nickel et al., 2011) that models the relation between two entities as a bilinear interaction $\langle \mathbf{h}, \mathbf{r}, \mathbf{t} \rangle$. Many other works have been proposed to enhance the translational and bilinear models such as modeling relational mapping properties (e.g., one-to-many and many-to-many) (Wang et al., 2014), modeling inference patterns (e.g., symmetry and composition) (Trouillon et al., 2016; Sun et al., 2019), and modeling complex graph structures (e.g., hierarchies and cycles) (Chami et al., 2020; Xiong et al., 2022d) to name a few.

Geometric relational embeddings Our work is closely related to geometric relational embeddings. See (Xiong et al., 2023) for a systematic survey. Geometric relational embeddings encode real-world relational knowledge by geometric objects such as convex regions like n -balls (Kulmanov et al., 2019), convex cones (Zhang et al., 2021; He et al., 2023), axis-parallel boxes (Vilnis et al., 2018; Xiong et al., 2022c; Ren et al., 2020) and non-Euclidean manifold components (Xiong et al., 2022a). A key advantage of these geometric embeddings is that they nicely model the set-theoretic semantics that can be used to capture logical rules of KGs (Abboud et al., 2020), ontological axioms (Kulmanov et al., 2019; Xiong et al., 2022c), transitive closure (Vilnis et al., 2018), and logical query for multi-hop reasoning (Ren et al., 2020). Different from all previous work, ShrinkE is the first geometric embedding that aims at modeling inference patterns for hyper-relational KGs.

B Proof of propositions

Proposition B.1. *Given any two facts $\mathcal{F}_1 = (\mathcal{T}, \mathcal{Q}_1)$ and $\mathcal{F}_2 = (\mathcal{T}, \mathcal{Q}_2)$ where $\mathcal{Q}_2 \subseteq \mathcal{Q}_1$, i.e., \mathcal{F}_2 is a partial fact of \mathcal{F}_1 , the output of the scoring function $f(\cdot)$ of ShrinkE satisfy the constraint $f(\mathcal{F}_2) \geq f(\mathcal{F}_1)$, which implies Eq.(2).*

Proof. We first prove that the resulting box of \mathcal{F}_2 subsumes the resulting box of \mathcal{F}_1 . Since the primal triple of \mathcal{F}_1 and \mathcal{F}_2 are the same (let assume it is $\mathcal{T} = (h, r, t)$), the spanned boxes of the two facts are $\mathcal{H}_r(\mathbf{e}_h)$. Since $\mathcal{Q}_2 \subseteq \mathcal{Q}_1$, the final shrunken box of \mathcal{F}_1 must be a subset of the shrunken box of \mathcal{F}_2 . Hence, we have,

$$\text{Box}_{\mathcal{F}_2} \subseteq \text{Box}_{\mathcal{F}_1}. \quad (15)$$

Given the tail entity t whose embedding is denoted by \mathbf{e}_t , we consider three cases of its position.

1) If \mathbf{e}_t is inside the small box $\text{Box}_{\mathcal{F}_2}$, then \mathbf{e}_t must also be inside $\text{Box}_{\mathcal{F}_1}$ since $\text{Box}_{\mathcal{F}_2} \subseteq \text{Box}_{\mathcal{F}_1}$. Note that our point-to-box function is monotonically increasing w.r.t. the increase of distance from the tail point to the center of box. Hence, we will have $D(\mathbf{e}, \text{Box}_{\mathcal{F}_2}) \geq D(\mathbf{e}, \text{Box}_{\mathcal{F}_1})$, implying $f(\mathcal{F}_2) \geq f(\mathcal{F}_1)$.

2) If \mathbf{e}_t is outside the small box $\text{Box}_{\mathcal{F}_2}$ but inside in the larger $\text{Box}_{\mathcal{F}_1}$, according to the definition of the point-to-box distance function, we immediately have $D(\mathbf{e}, \text{Box}_{\mathcal{F}_2}) \geq D(\mathbf{e}, \text{Box}_{\mathcal{F}_1})$, implying $f(\mathcal{F}_2) \geq f(\mathcal{F}_1)$.

3) If \mathbf{e}_t is outside the larger box $\text{Box}_{\mathcal{F}_1}$, then \mathbf{e}_t must also be outside $\text{Box}_{\mathcal{F}_2}$ since $\text{Box}_{\mathcal{F}_2} \subseteq \text{Box}_{\mathcal{F}_1}$. Note that our point-to-box function is monotonically decreasing w.r.t. the increase of volume of box. Hence, we will have $D(\mathbf{e}, \text{Box}_{\mathcal{F}_2}) \geq D(\mathbf{e}, \text{Box}_{\mathcal{F}_1})$, implying $f(\mathcal{F}_2) \geq f(\mathcal{F}_1)$. \square

Proposition B.2. *ShrinkE is able to infer hyper-relational symmetry, anti-symmetry, inversion, composition, hierarchy, intersection, and exclusion.*

We first prove that ShrinkE is able to infer symmetry, anti-symmetry, inversion, and composition. For the sake of proof, we assume $\theta_r \in [-\pi, \pi)$. We prove them by proving Lemma B.1-4 one by one.

Lemma B.1 (Symmetry). *Let r be a symmetric relation such that for each triple (e_h, r, e_t) , its symmetric triple (e_t, r, e_h) also holds. This symmetric property of r can be modeled by ShrinkE.*

Proof. If r is a symmetric relation, by taking the $\delta_r = \mathbf{0}$, $\mathbf{b}_r = \mathbf{0}$, and $\Theta_r = \text{diag}(\mathbf{G}(\theta_{r,1}), \dots, \mathbf{G}(\theta_{r,\frac{d}{2}}))$, where $\mathbf{G}(\theta)$ is a 2×2 diagonal matrix, we have

$$\begin{aligned} \mathbf{e}_h &= f_r(\mathbf{e}_t) = \Theta_r \mathbf{e}_t, \quad \mathbf{e}_t = f_r(\mathbf{e}_h) = \Theta_r \mathbf{e}_h \\ &\Rightarrow \Theta_r^2 = \mathbf{I} \end{aligned}$$

which holds true when $\theta_{r,i} = \mathbf{0}$ or $\theta_{r,i} = -\pi$ for $i = 1, \dots, \frac{d}{2}$. \square

Lemma B.2 (Anti-symmetry). *Let r be an anti-symmetric relation such that for each triple (e_h, r, e_t) , its symmetric triple (e_t, r, e_h) is not true. This anti-symmetric property of r can be modeled by ShrinkE.*

Proof. If r is a anti-symmetric relation, by taking the $\delta_r = \mathbf{0}$, $\mathbf{b}_r = \mathbf{0}$, and $\Theta_r = \text{diag}\left(\mathbf{G}(\theta_{r,1}), \dots, \mathbf{G}\left(\theta_{r, \frac{d}{2}}\right)\right)$, where $\mathbf{G}(\theta)$ is a 2×2 diagonal matrix, we have

$$\begin{aligned} e_h \neq f_r(e_t) &= \Theta_r e_t, \quad e_t = f_r(e_h) = \Theta_r e_h \\ &\Rightarrow \Theta_r^2 \neq \mathbf{I} \end{aligned}$$

which holds true when $\theta_{r,i} \neq \mathbf{0}$ or $\theta_{r,i} \neq -\pi$ for $i = 1, \dots, \frac{d}{2}$. \square

Lemma B.3 (Inversion). *Let r_1 and r_2 be inverse relations such that for each triple (e_h, r_1, e_t) , its inverse triple (e_t, r_2, e_h) is also true. This inverse property of r_1 and r_2 can be modeled by ShrinkE.*

Proof. If r_1 and r_2 are inverse relations, by taking the $\delta_r = \mathbf{0}$, $\mathbf{b}_r = \mathbf{0}$, and $\Theta_r = \text{diag}\left(\mathbf{G}(\theta_{r,1}), \dots, \mathbf{G}\left(\theta_{r, \frac{d}{2}}\right)\right)$, where $\mathbf{G}(\theta)$ is a 2×2 diagonal matrix, we have

$$\begin{aligned} e_t = f_{r_1}(e_h) &= \Theta_{r_1} e_h, \quad e_h = f_{r_2}(e_t) = \Theta_{r_2} e_h \\ &\Rightarrow \Theta_{r_1} \Theta_{r_2} = \mathbf{I} \end{aligned}$$

which holds true when for $\theta_{r_1,i} r_1 + \theta_{r_2,i} = 0$ for $i = 1, \dots, \frac{d}{2}$. \square

Lemma B.4 (Composition). *Let relation r_1 be composed of r_2 and r_3 such that triple (e_1, r_1, e_3) exists when (e_1, r_2, e_2) and (e_2, r_3, e_3) exist. This composition property can be modeled by ShrinkE.*

Proof. If r_1 is composed of r_2 and r_3 , by taking the $\delta_r = \mathbf{0}$, $\mathbf{b}_r = \mathbf{0}$, and $\Theta_r = \text{diag}\left(\mathbf{G}(\theta_{r,1}), \dots, \mathbf{G}\left(\theta_{r, \frac{d}{2}}\right)\right)$, where $\mathbf{G}(\theta)$ is a 2×2 diagonal matrix, we have

$$\begin{aligned} e_3 = f_{r_1}(e_1) &= \Theta_{r_1} e_1, \quad e_2 = f_{r_2}(e_1) = \Theta_{r_2} e_1, \\ e_3 = f_{r_3}(e_2) &= \Theta_{r_3} e_2 \Rightarrow \Theta_{r_1} = \Theta_{r_2} \Theta_{r_3} \end{aligned}$$

which holds true when $\theta_{r_1,i} = \theta_{r_2,i} + \theta_{r_3,i}$ or $\theta_{r_1,i} = \theta_{r_2,i} + \theta_{r_3,i} + 2\pi$ or $\theta_{r_1,i} = \theta_{r_2,i} + \theta_{r_3,i} - 2\pi$ for $i = 1, \dots, \frac{d}{2}$. \square

We now prove that ShrinkE is able to infer relation implication, exclusion and intersection.

Lemma B.5 (Relation implication). *Let $r_1 \rightarrow r_2$ form a hierarchy such that for each triple (e_h, r_1, e_t) , (e_h, r_2, e_t) also holds. This hierarchy property $r_1 \rightarrow r_2$ can be modeled by ShrinkE.*

Proof. If $r_1 \rightarrow r_2$, by taking $\mathcal{T}_{r_1} = \mathcal{T}_{r_2}$, i.e., $\delta_{r_1} = \delta_{r_2}$ and $\Theta_{r_1} = \Theta_{r_2}$, we have, $(e_h, r_1, e_t) \rightarrow (e_h, r_2, e_t)$ implies that the spanning box of query $(e_h, r_1, x?)$ is subsumed by the spanning box of query $(e_h, r_2, x?)$. i.e., $\text{Box}(\mathcal{H}_{r_1}(e_h) - \sigma(\delta_{r_1}), \mathcal{H}_{r_1}(e_h) + \sigma(\delta_{r_1})) \subseteq \text{Box}(\mathcal{H}_{r_1}(e_h) - \sigma(\delta_{r_2}), \mathcal{H}_{r_1}(e_h) + \sigma(\delta_{r_2}))$, which holds true when $\delta_{r_1} \leq \delta_{r_2}$. \square

Lemma B.6 (Relation exclusion). *Let r_1, r_2 be mutually exclusive, that is, (e_h, r_1, e_t) , (e_h, r_2, e_t) can not be simultaneously hold. This mutual exclusion property $r_1 \wedge r_2 \rightarrow \perp$ can be modeled by ShrinkE.*

Proof. If $r_1 \wedge r_2 \rightarrow \perp$, we have $(e_h, r_1, e_t) \wedge (e_h, r_2, e_t) \rightarrow \perp$, which implies that the spanning box of query $(e_h, r_1, x?)$ and the spanning box of query $(e_h, r_2, x?)$ are mutually exclusive, i.e., $\text{Box}(\mathcal{H}_{r_1}(e_h) - \sigma(\delta_{r_1}), \mathcal{H}_{r_1}(e_h) + \sigma(\delta_{r_1})) \cap \text{Box}(\mathcal{H}_{r_1}(e_h) - \sigma(\delta_{r_2}), \mathcal{H}_{r_1}(e_h) + \sigma(\delta_{r_2})) \rightarrow \perp$. \square

Lemma B.7 (Relation intersection). *Let r_3 be a intersection of r_1, r_2 , that is, if (e_h, r_1, e_t) and (e_h, r_2, e_t) hold, then (e_h, r_3, e_t) also holds. This intersection property $r_1 \wedge r_2 \rightarrow r_3$ can be modeled by ShrinkE.*

Proof. Note that box is closed under intersection and this property can be view as a combination of two pairs of relation implication. Hence, the proof is similar to the proof of Lemma B. \square

Proposition B.3. *ShrinkE is able to infer qualifier implication, mutual exclusion, and intersection.*

Proof. Since each qualifier is associated with a box, the implication and mutual exclusion relationships between qualifiers can be modeled by their geometric relationships, i.e., box entailment and box disjointedness, respectively, between their corresponding boxes. Qualifier intersection can be modeled by enforcing the box of one qualifier to be inside the intersection of the boxes of another two qualifiers. \square

ACL 2023 Responsible NLP Checklist

A For every submission:

- A1. Did you describe the limitations of your work?
Limitation section
- A2. Did you discuss any potential risks of your work?
In "Ethics Statement"
- A3. Do the abstract and introduction summarize the paper's main claims?
Abstract
- A4. Have you used AI writing assistants when working on this paper?
Left blank.

B Did you use or create scientific artifacts?

Left blank.

- B1. Did you cite the creators of artifacts you used?
No response.
- B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.
- B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
No response.
- B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No response.
- B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.
- B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
No response.

C Did you run computational experiments?

Section 6

- C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix C.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

- C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 6 and Appendix C.
- C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Left blank.
- C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Left blank.
- D** **Did you use human annotators (e.g., crowdworkers) or research with human participants?**
Left blank.
- D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.
- D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.
- D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.
- D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.
- D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.