Data Mining and Knowledge Discovery (2024) 38:3531–3562
https://doi.org/10.1007/s10618-024-01050-x

Knowledge graph embedding closed under composition

Zhuoxun Zheng[1,2] · Baifan Zhou[2,3] · Hui Yang[4] · Zhipeng Tan[1] · Zequn Sun[5] · Chunnong Li[6] · Arild Waaler[2] · Evgeny Kharlamov[1,2] · Ahmet Soylu[2,3]

Received: 6 December 2023 / Accepted: 12 June 2024 / Published online: 4 July 2024

Abstract

Knowledge Graph Embedding (KGE) has attracted increasing attention. Relation patterns, such as symmetry and inversion, have received considerable focus. Among them, composition patterns are particularly important, as they involve nearly all relations in KGs. However, prior KGE approaches often consider relations to be compositional only if they are well-represented in the training data. Consequently, it can lead to performance degradation, especially for under-represented composition patterns. To this end, we propose HolmE, a general form of KGE with its relation embedding space closed under composition, namely that the composition of any two given relation embeddings remains within the embedding space. This property ensures that every relation embedding can compose, or be composed by other relation embeddings. It enhances HolmE's capability to model under-represented (also called long-tail) composition patterns with limited learning instances. To our best knowledge, our work is pioneering in discussing KGE with this property of being closed under composition. We provide detailed theoretical proof and extensive experiments to demonstrate the notable advantages of HolmE in modelling composition patterns, particularly for long-tail patterns. Our results also highlight HolmE's effectiveness in extrapolating to unseen relations through composition and its state-of-the-art performance on benchmark datasets.

Keywords Knowledge graph embedding · Link prediction · Machine learning

Responsible editor: Panagiotis Papapetrou.

✉ Zhuoxun Zheng
zhuoxun.zheng@de.bosch.com

[1] Bosch Center for AI, Renningen, Germany

[2] University of Oslo, Oslo, Norway

[3] Oslo Metropolitan University, Oslo, Norway

[4] University of Paris-Saclay, Orsay, France

[5] Nanjing University, Nanjing, China

[6] Ludwig Maximilian University of Munich, Munich, Germany

1 Introduction

Knowledge graphs (KGs) refer to multi-relational graphs that represent facts commonly in the form of triples. Research in KG embedding (KGE) aims to encode KGs into numerical representations, enabling the utilisation of powerful machine learning techniques for various downstream tasks, including link prediction (Rossi et al. 2021), query answering (Arakelyan et al. 2020), entity classification (Wang et al. 2017), and more. Additionally, certain KGE models have identified the effectiveness of modelling KGs in hyperbolic space, which allows them to represent KGs more efficiently in lower dimensions (Balažević et al. 2019).

An essential focus of KGE methods for link prediction is the modelling of diverse relation patterns, such as symmetry/asymmetry, inversion, composition, etc. Sun et al. (2019), Abboud et al. (2020), Bai et al. (2021). Among these relation patterns, composition patterns are particularly important, as almost all relations participate in some composition patterns: they can compose or be composed by other relations (see Table 1 for established benchmark KG datasets). Indeed, composition patterns enable logical deductions and reasoning (Datta et al. 2007), which are natural ways humans acquire and deduce new knowledge. In the real world, a substantial portion of link prediction queries can be derived through composition patterns.

However, past works often model relations as compositional solely based on the training data. This effectively assumes that the relations are non-compositional if their associated composition patterns are not well-represented in the training data. However, this restrictive assumption poses a theoretical contradiction to real-world relations and can be problematic in practice. If a relation is only associated with under-represented (also known as long-tail) composition patterns, its representation tends to be embedded in a sub-space where the composition patterns are not supported (Fig. 1a). Consequently, this restrictive assumption can result in performance degradation when modelling long-tail composition patterns with limited learning instances, which constitute a significant majority within

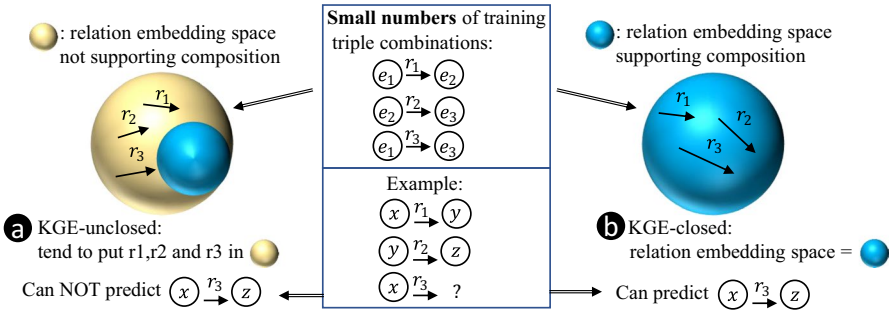

Fig. 1 KGE-closed has the advantage that the entire relation embedding space supports composition, so that even under-represented composition patterns ($r_1 \circ r_2 \rightarrow r_3$) can be properly learned. In contrast, KGE-unclosed tends to learn under-represented composition patterns in the relation embedding space that does not support composition

benchmark KG datasets (see Fig. 3). Moreover, the future of KGE can anticipate scenarios with vast KG data and continuous learning wherein certain relations initially not considered compositional in the original dataset may become involved in composition patterns as new training triples emerge. In such cases, KGEs with this restrictive assumption will struggle to learn the correct representation of those relations, as their evolving relation patterns affect the essential dynamics between relations and entities. Therefore, a more effective approach in KGE would be to not exclude the potential of any two relations composing a new relation.

To this end, we propose HolmE, a general form of Riemannian KGE with the relation embedding space closed under composition, which means the composition of any two given relation embeddings remains within the same embedding space (Fig. 1b). As a result, HolmE offers several benefits: (1) HolmE is theoretically elegant as it gives the property that aligns with the nature of real-world relations, in the sense that it allows every relation embedding to compose or be composed by other relation embeddings; (2) it empowers the model with advanced capabilities to model composition patterns, even in long-tail scenarios where these patterns are under-represented in the training data; (3) the closure under composition enhances HolmE's capability to extrapolate to unseen relations based on composition patterns, strengthening its predictive capability further. To our best knowledge, our work is pioneering in discussing KGE with the property of being closed under composition.

This journal paper builds upon our conference paper (Zheng et al. 2024) with several significant extensions: we provide an in-depth analysis of the representation of composition patterns within the KG data and quantify them with numerical metrics (Sect. 3.4); we delve deeply into the data to demonstrate that composition patterns constitute significant components in benchmark KG datasets (Sect. 5.1); moreover, we conduct experiments on extra datasets and verify three research hypotheses for demonstrating the superior properties of HolmE in capturing composition patterns, especially in the long-tail scenarios (Sects. 5.2–5.3). We summarise the contributions of this paper as below:

- We introduce the property of KGE models as being closed under composition. We also demonstrate the significance of composition patterns by showing a major part of relations participate in various composition patterns.
- We propose HolmE, a general form of KGE models in Riemannian space, and provide detailed theoretical proof and empirical evaluation to illustrate the property of being closed under composition. We also theoretically prove that TransE (Bordes et al. 2013) and RotatE (Sun et al. 2019) are special cases of HolmE.
- We conduct extensive experiments that demonstrate HolmE's advantages in representing composition patterns, particularly in scenarios with a restricted number of learning instances. Additionally, we demonstrate HolmE's effectiveness in extrapolating to unseen relations based on composition patterns.
- We show that HolmE has state-of-the-art performance on benchmark datasets.

The code and data are open-sourced under https://github.com/nsai-uio/HolmE-KGE.

2 Related work

Traditional KGE models for link prediction. The link prediction task aims to infer new triples by using existing ones in KGs. Traditionally, there are two groups of KGE models intended for such task, namely geometric models and bilinear models. Geometric KGE models employ expressive embeddings for entities and effective mathematical mappings for relations, offering clear geometric insights into relation patterns with efficient resource utilization. Geometric KGE models are pursued along two dimensions: entity embedding spaces and mathematical mappings for relations. Notably, TransE (Bordes et al. 2013) employs Euclidean space for entities with translation for relations, effectively capturing various relation patterns while maintaining simplicity. To empower TransE for more capability for representing such as 1-N relations, subsequent works, including TransH (Wang et al. 2014), TransR (Lin et al. 2015), and TransD (Ji et al. 2015), introduced projection as a new kind of relation mapping. RotatE (Sun et al. 2019) and its generalizations, RotatE3d (Gao et al. 2020) and QuatE (Zhang et al. 2019) introduced rotation for relation mappings to represent more relation patterns, such as symmetry. Besides, hyperbolic space has been investigated and utilized for entity embeddings, which achieve high performance with low embedding dimensions due to its high expressivity and suitability for hierarchical structure. These works include MurP (Balažević et al. 2019), AttH (Chami et al. 2020), and GIE (Cao et al. 2022). Another group of KGE models embed both entities and relations as matrices and perform multiplication scoring, such as RESCAL (Nickel et al. 2011), which is prone to overfitting due to the large number of embedding parameters. DistMult (Yang et al. 2014) addressed the overfitting problem but failed to model asymmetric patterns. Later, ComplEx (Trouillon et al. 2016) achieved state-of-the-art performance equipped with the N3 regularization method (Lacroix et al. 2018).

Neural networks-based KGE models. Neural network approaches such as Jin et al. (2021) and Yu et al. (2021) play important roles in graph learning. In the domain of knowledge graph representation, characterised by the presence of heterogeneous relations, numerous cutting-edge neural network-based approaches have been introduced. These methods have demonstrated considerable success in link prediction and other downstream tasks. Examples of such models for link prediction task include R-GCN (Schlichtkrull et al. 2018), ConvE (Dettmers et al. 2018), SE-GNN (Li et al. 2022), and KBGAT (Nathani et al. 2019), which employ convolutional or attention-based neural networks to capture intricate patterns and interdependencies within the knowledge graphs. These methods provide sophisticated approaches to represent intricate semantic relationships in a continuous vector space, offering valuable insights into knowledge graph structures and interconnections. However, they often require significant time and resources and may lack explicit explanations for modelling relation patterns (Pavlović and Sallinger 2022).

KGE in modelling long-tail items. Traditional link prediction models often require a substantial number of labelled triples for effective training, limiting their applicability in modelling long-tail entities or relations. The emergence of few-shot knowledge graph completion addresses this limitation by focusing on models capable of making accurate predictions with minimal training instances. Researchers in this domain aim to develop models that can effectively represent entities and relations that are less represented during training (Zhang et al. 2020). LAN (Wang et al. 2019), GMatching (Xiong et al. 2018) utilize neighborhood aggregation through neural networks to embed unseen or long-tail entities, which are connected with the training KG. GraIL (Teru et al. 2020), MorsE (Chen et al. 2022), and NBFNet (Zhu et al. 2021) capture entity-independent meta-knowledge from the training KG, enabling generalization to new KGs containing unseen entities.

Our work primarily delves into the mathematical properties inherent in traditional KGEs that explicitly capture relation patterns. Consequently, we omit discussion on neural network-based models. The key mathematical property we emphasise, namely closure under composition, enhances our model's capacity to effectively represent long-tail composition patterns. It's essential to note that our approach differs from typical few-shot learning KGEs, as we specifically focus on composition patterns with minimal representation in the training data rather than on entities or relations themselves.

3 Problem setup

3.1 Knowledge graph embedding (KGE)

A knowledge graph (KG) is a structured representation of information in the form of a graph. Formally, a knowledge graph (KG) is a multi-relational graph $G = (\mathcal{E}, \mathcal{R}, \mathcal{F})$, where $\mathcal{E}$ is a set of entities (nodes), $\mathcal{R}$ is a set of binary relations (edge types) between entities, and $\mathcal{F}$ is a set of facts (edges) given in the triple form of (h, r, t), with h, t and r denoting the head entity, tail entity, and the relation in between, respectively.

A knowledge graph embedding (KGE) model is used to map the entities and relations in a KG into numerical representations (named as embeddings) in a continuous vector space, named as the embedding space. In the common link prediction setting of KGE problems $(h, r, ?)$ (or $(?, r, t)$), where the *query* of the head entity (or tail entity) and the relation are given, we want to find the most probable tail entity (or head entity). The embeddings are learned in such a way that a scoring function $f(h, r, t)$ assigns higher scores to valid triples $(h, r, t) \in \mathcal{F}$ and lower scores to invalid triples. For simplicity, the query in both directions is denoted as $(h, r) \rightarrow t$.

Formally, let $(\mathcal{P}, \mathcal{G})$ be an embedding space, where $\mathcal{P}$ is a distance space, $\mathcal{G}$ is a set of mappings that have domain and range defined on $\mathcal{P}$, and s is a scoring function: $s : \mathcal{P} \times \mathcal{G} \times \mathcal{P} \rightarrow \mathbb{R}$. The KGE problem aims to find an embedding from G to

$(\mathcal{P}, \mathcal{G})$ that (i) maps entities $h, t \in \mathcal{E}$ to points (vectors) $\mathbf{e}_h, \mathbf{e}_t \in \mathcal{P}$; (ii) maps each relation $r \in \mathcal{R}$ to a mapping $g_r \in \mathcal{G}$ such that $s(\mathbf{e}_h, g_r, \mathbf{e}_t)$ ranks how probable that $(h, r, t) \in \mathcal{F}$.

3.2 Riemannian geometry

We briefly introduce Riemannian geometry and provide more information in Appendix D. A Riemannian manifold is a generalisation of Euclidean space with a flexible curvature. Curvature is a measure for calculating the distance between points/vectors in a geometric space (Willmore 2013). This paper focuses on the hyperbolic space, which is a type of Riemannian manifold with constant negative curvature $-\kappa$ and dimension d, represented as $\mathbb{H}^{d,\kappa} := \{\mathbf{x} \in \mathbb{R}^d \mid ||\mathbf{x}||^2 < \frac{1}{\kappa}\}$, where $||\cdot||$ denotes the L2 norm. Intuitively, considering the surface area of a hypersphere with an increasing radius centred at a certain point, Euclidean space can be seen to have polynomial "growth", while in hyperbolic space, the analogous expansion showcases an exponential increase (Sala et al. 2018). Thus, the distance between points increases exponentially as they move away from the origin in hyperbolic space than in Euclidean space. This property of hyperbolic space allows for more efficient organisation of hierarchical structures, such as trees or networks (Balažević et al. 2019).

Analogous to the addition in Euclidean space, the translation in hyperbolic space is defined as the Möbius addition (Ganea et al. 2018), denoted as $\oplus^\kappa$:

$$\mathbf{x} \oplus^\kappa \mathbf{y} = \frac{(1 + 2\kappa\langle \mathbf{x}, \mathbf{y}\rangle + \kappa||\mathbf{y}||^2)\mathbf{x} + (1 - \kappa||\mathbf{x}||^2)\mathbf{y}}{1 + 2\kappa\langle \mathbf{x}, \mathbf{y}\rangle + \kappa^2||\mathbf{x}||^2||\mathbf{y}||^2},$$

where $\mathbf{x}$ and $\mathbf{y}$ are the two vectors to be "added" in the hyperbolic space, $\langle \mathbf{x}, \mathbf{y}\rangle$ is the inner product of the two vectors, and κ is the curvature. Based on this, the distance between $\mathbf{x}$ and $\mathbf{y}$ is defined as: $d^\kappa(\mathbf{x}, \mathbf{y}) = 2/\sqrt{\kappa} \cdot arctanh(\sqrt{\kappa}||-\mathbf{x} \oplus^\kappa \mathbf{y}||)$.

Product space of hyperbolic space: A product space is the Cartesian product of multiple spaces. Here we denote the product space $\mathbb{P}_{m,n,\kappa}$ of dimension $d = m \cdot n$, which is the Cartesian product of m hyperbolic spaces $\mathbb{H}^{n,\kappa}$ of dimension n. A vector on $\mathbb{P}_{m,n,\kappa}$ can be decomposed into m sub-vectors in $\mathbb{H}^{n,\kappa}$.

We can also extend the Möbius addition on hyperbolic space to the product space of hyperbolic spaces. This is more computationally efficient than the computation of Möbius addition on a high-dimensional hyperbolic space. For any $\mathbf{x}, \mathbf{y} \in \mathbb{P}_{m,n,\kappa}$, where $\mathbf{x} = (\mathbf{x}^1, \ldots, \mathbf{x}^m)$, $\mathbf{y} = (\mathbf{y}^1, \ldots, \mathbf{y}^m)$, the Möbius addition $\oplus^\kappa$ can be extended to $\oplus^\kappa_{\mathcal{P}}$ on $\mathbb{P}_{m,n,\kappa}$:

$$\mathbf{x} \oplus^\kappa_{\mathcal{P}} \mathbf{y} = (\mathbf{x}^1 \oplus^\kappa \mathbf{y}^1, \ldots, \mathbf{x}^m \oplus^\kappa \mathbf{y}^m). \tag{1}$$

The distance between $\mathbf{x}, \mathbf{y}$ is calculated as the sum of all Riemannian distances between the sub-vectors $\mathbf{x}^i, \mathbf{y}^i$ (Eq. 2).

$$d^\kappa_{\mathcal{P}}(\mathbf{x}, \mathbf{y})^2 = \sum_{i=1}^{m} \left(\frac{2}{\sqrt{\kappa}} arctanh(\sqrt{\kappa}||-\mathbf{x}^i \oplus^\kappa \mathbf{y}^i||) \right)^2. \tag{2}$$

3.3 Modelling relation patterns

The key of link prediction in KG is to infer the connection patterns, e.g., relation patterns (Sun et al. 2019). We list some relation patterns in the following according to past works (Trouillon et al. 2016).

Symmetry/Asymmetry: Relation r is symmetric if triple (x, r, y) holds, then (y, r, x) also holds, formally: $\forall x, y, \quad r(x, y) \Rightarrow r(y, x)$. On the contrary, r is asymmetric if triple (x, r, y) holds, then (y, r, x) is invalid, formally: $\forall x, y, \quad r(x, y) \Rightarrow \neg r(y, x)$.

Inversion: Relation r_1 is inverse to relation r_2, if triple (x, r_1, y) is valid, then (y, r_2, x) is valid, formally: $\forall x, y, \quad r_2(x, y) \Rightarrow r_1(y, x)$.

Transition: Relation r is transitive, if triples (x, r, y) and (y, r, z) are valid, then triple (x, r, z) is valid, formally: $\forall x, y, z, \quad r(x, y), r(y, z) \Rightarrow r(x, z)$.

We can observe that these relation patterns can only apply to certain relations, not all, namely: it is for sure that some relations are not transitive; a relation cannot be symmetric and asymmetric simultaneously; not all relation pairs can form inverse patterns. Given this, the composition pattern is different, explained below.

3.4 Modelling composition patterns

The composition pattern is broader in scope because it allows any relation to be involved. In theory, any two relations can combine to create a new relation. Additionally, the transition pattern can be seen as a specific type of composition that involves only one identical relation. The composition pattern is defined as follows Sun et al. (2019).

Composition Pattern: Relation r_3 is composed of relation r_1 and r_2, denoted as $r_3 = r_1 \circ r_2$, iff $\forall x, y, z$, we have $r_1(x, y), r_2(y, z) \Rightarrow r_3(x, z)$.

From a data-centric standpoint, we define the triple count and the representing ratio for a composition pattern below, which indicate how robustly a composition pattern is represented in the data.

Definition 1 *Triple count of a composition pattern* ($\mathbf{n}_{CP}$): *The triple count of a certain composition pattern is the number of the triples that form that pattern.*

Definition 2 *Representing ratio of a composition pattern* ($\mathbf{r}_{CP}$): *The representing ratio of a composition pattern is the ratio between its triple count* $\mathbf{n}_{CP}$ *and the total number of triples.*

For example for dataset $\{(a, r_1, b), (a, r_1, c), (b, r_2, d), (c, r_2, d), (a, r_3, d)\}$, $\mathbf{n}_{r_1 \circ r_2 = r_3}$ (the triple count for the composition pattern $r_1 \circ r_2 = r_3$) is 5, and $\mathbf{r}_{r_1 \circ r_2 = r_3}$ is 100%. Because all the 5 triples in the dataset participate in this pattern through two paths: $((a, r_1, b), (b, r_2, d), (a, r_3, d))$ and $((a, r_1, c), (c, r_2, d), (a, r_3, d))$.

Intuitively, a composition pattern is acknowledged only when its $\mathbf{r}_{CP}$ (or $\mathbf{n}_{CP}$) is over some threshold, as this ensures the pattern is recognised with confidence rather

than being dismissed as a mere coincidence. For instance, considering the triples {(a, *hasFather*, b), (b, *hasMother*, c), (a, *hasTeacher*, c)}, attempting to derive the pattern $hasFather \circ hasMother = hasTeacher$ is inappropriate. This combination of relations appears coincidental rather than constituting a meaningful pattern, as the triples forming this combination must be rare within the dataset, which is illustrated by an $\mathbf{r}_{CP}$ (or $\mathbf{n}_{CP}$) below a certain threshold. In contrast, the greater the $\mathbf{r}_{CP}$ is (or the greater the $\mathbf{n}_{CP}$ within an identical KG dataset is), the more robustly the pattern is deemed to be represented in the data. A small $\mathbf{r}_{CP}$ (or $\mathbf{n}_{CP}$) indicates a long-tail composition pattern that is less represented in the training data, which however constitutes a significant majority within benchmark KG datasets (see Fig. 3).

From a KGE-centric standpoint, each relation r is embedded into a mathematical mapping $g_r \in \mathcal{G}$. we use the notation $g_{r_2} \circ g_{r_1}$ to express the composition of the two mappings g_{r_1} and g_{r_2}, i.e., $g_{r_2} \circ g_{r_1}(\mathbf{e}_x) := g_{r_2}(g_{r_1}(\mathbf{e}_x))$, which we hope to be identical to the embedded mapping g_{r_3} for r_3, given $r_1 \circ r_2 = r_3$. Thus, the composed mapping $g_{r_2} \circ g_{r_1}$ should be confined within $\mathcal{G}$ and be aligned closed to g_{r_3}. Based on the assumption of real-world relations that any two relations can compose a new relation (as long as the connecting entity exists), we define the closed relation embedding space under composition as a fundamental property of KGE, which is defined as follows:

Definition 3 A KGE $(\mathcal{P}, \mathcal{G})$ is closed under composition *(KG-closed)*, iff for any $g_{r_1}, g_{r_2} \in \mathcal{G}$, we have $g_{r_1} \circ g_{r_2} \in \mathcal{G}$.

4 Method: HolmE

We aim to design a general KGE model that (1) is closed under composition, thereby enhancing its expressiveness for modelling composition patterns, especially in long-tail scenarios; and (2) can also encode various other relational patterns. We elaborate on the formulas of Holme and the rationale behind them, then prove Holme to be closed under composition. and then compare HolmE to representative KGE methods.

Intuition. HolmE decomposes the head entity embedding e_h (Fig. 2.0) in the product space $\mathbb{P}_{m,n,\kappa}$ to sub-vectors e_h^i (Fig. 2.1) in the component spaces, $i \in [1, m]$. Subsequently, relation-specific translations and rotations are performed on each sub-vector (Fig. 2.2); then, the sub-vectors are concatenated (Fig. 2.3) to the predicted embedding of the tail entity (Fig. 2.4). We give the general form of HolmE as follows:

Definition 4 Let $(\mathcal{P}, \mathcal{G})$ be the embedding space, where entity embedding space $\mathcal{P} = \mathbb{P}_{m,n,\kappa}$, and relation embedding space $\mathcal{G}$ is a group of mappings of the form:

$$g_r(\mathbf{e}_h) = \mathbf{a}_r \oplus_{\mathcal{P}}^{\kappa} (\mathbf{A}_r \cdot \mathbf{e}_h). \tag{3}$$

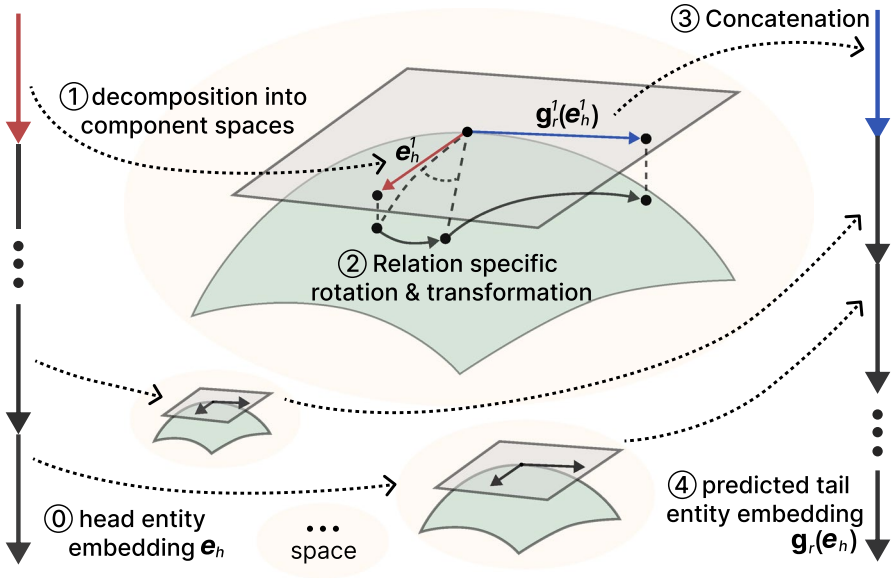

Fig. 2 To predict tail embedding of (h,r,?), HolmE decomposes head entity embedding e_h in manifold $\mathbb{P}_{m,n,\kappa}$ (0) to sub-vectors e_h^i in the i-th component space, $i \in [1, m]$ (1). Subsequently relation-specific translations and rotations g_r^i are conducted on each sub-vector (2); then, the sub-vectors are concatenated (3) to the predicted embedding of the tail entity $g_r(\mathbf{e}_h)$(4)

with parameters of relation-specific translation, $\mathbf{a}_r \in \mathcal{P}_{d,n}$ and rotation $\mathbf{A}_r \in (\mathbf{SO}(n))^m$ (m-times production special orthogonal matrix $\mathbf{SO}(n)$); $\oplus_{\mathcal{P}}^{\kappa}$ is the extended Möbius addition (defined in Eq. 1) on product space $\mathcal{P}$ with curvature κ.

The scoring function s of a tuple (h, r, t) is defined by

$$s(\mathbf{e}_h, g_r, \mathbf{e}_t) = -d_{\mathcal{P}}^{\kappa}\big(g_r(\mathbf{e}_h), \mathbf{e}_t\big) \tag{4}$$

where $\mathbf{e}_h, \mathbf{e}_t \in \mathcal{P}$ are head and tail entity embeddings; the distance $d_{\mathcal{P}}^{\kappa}$ is calculated on the product space $\mathcal{P}$ (defined in Eq. 2).

We now elaborate on each part of HolmE and the rationales behind them.

Entity embedding. Each entity is embedded as a $m * n$-dimensional vector in product space $\mathbb{P}_{m,n,\kappa}$ with a learned constant curvature κ (Fig. 2.0), it is decomposed into m sub-vectors, each sub-vector is in a hyperbolic space $\mathbb{H}^{n,\kappa}$ with the dimension n (Fig. 2.1). These component spaces also share the same κ. The curvature is learned because it adjusts the embedding space to better distribute points throughout the space (Chami et al. 2020). We adopt a constant curvature because it is required for the closed space under composition (Appendix A).

Relation embedding. The relation mapping g_r consists of the rotation in product space and the translation in product space, which are introduced in the following:

Rotation in the product space. A tempting choice is to model rotation simply as a high dimensional rotation matrix, which contains a high number of parameters and is very difficult to learn.

HolmE breaks down the high-dimensional rotation into a series of n-dimensional rotations on the m component spaces (Fig. 2.2). The n dimensional rotation matrix $\mathbf{A}_r$ is a special orthogonal matrix (namely $|\mathbf{A}_r| = 1$), in the form of a diagonal rotation matrix consisting of a series of rotation matrices of n dimensions: $\mathbf{A}_r = diag[\mathbf{R}^n(\theta_{r,1}), \ldots, \mathbf{R}^n(\theta_{r,m})]$, where $\mathbf{R}^n(\theta_r)$ is a n dimensional rotation matrix, $(n = 2, 3, \ldots)$. When $n = 2$, $\mathbf{R}^n(\theta_r)$ is the special case of rotation in the complex plane (Eq. 5). Moreover, if $m = \kappa = 1$, the mappings of HolmE are equivalent to *holomorphic functions* over the unit disk of the complex space $\mathbb{C}$ (Ungar 2001) (this is where the name HolmE comes from). For higher-dimensional $\mathbf{R}^n(\theta_r)$ see Appendix E.

$$\mathbf{R}^n(\theta_r) := \begin{bmatrix} cos(\theta_r) & -sin(\theta_r) \\ sin(\theta_r) & cos(\theta_r) \end{bmatrix}. \tag{5}$$

Translation in the product space. The translation of HolmE is performed with extended *Möbius addition* (Eq. 1) in the component spaces with curvature κ (Fig. 2.2). This means the Möbius addition is performed between the n dimensional sub-vectors of $\mathbf{a}_r$ and the resulting vectors of the rotation $(\mathbf{A}_r \cdot \mathbf{e}_h)$, where each sub-vector belongs to a component $\mathbb{H}^{n,\kappa}$ of $\mathbb{P}_{m,n,\kappa}$. We adopt the translation in the product space instead of high-dimensional translation, different from past work (Chami et al. 2020), because translation in the product space matches the rotation in the product space and thus makes more sense geometrically, which also leads to the property of closure under composition Important to note is that the Möbius addition here is the *left addition*, namely the translation vector $\mathbf{a}_r$ must be on the left hand side, for ensuring a closed embedding space under composition (proof see Appendix C).

Scoring and loss function. The scoring function s (Eq. 4) is designed in the product space instead of in the high-dimensional manifold because it matches the rotation and translation in the product space and thus makes more sense geometrically, which also leads to a closed relation embedding space under composition. The loss function $\mathcal{L}$ is adopted as the cross-entropy loss with uniform negative sampling:

$$\mathcal{L} = \sum_{(h,r,t)\in\mathcal{F}\cup\mathcal{F}\simeq} log(1 + exp(-y \cdot s(\mathbf{e}_h, g_r, \mathbf{e}_t))), \tag{6}$$

where $\mathcal{F}$ and $\mathcal{F}\simeq$ denote training examples and negative examples, respectively. The negative examples are sampled uniformly from all possible triplets obtained by perturbing the tail entities in triplets (h, r, t). The label y is 1 if the triplet is a training example; otherwise it is -1.

Training and optimisation The training setting follows the initialisation in tangent spaces as Chami et al. (2019). In particular, all parameters are initialised and optimised in the tangent space of the manifold using standard Euclidean techniques. These parameters are mapped to the manifold with exponential map depending on the κ, which ensures the embeddings are in the desired space (Chami et al. 2019).

We follow the standard evaluation protocol in the filtered setting (Bordes et al. 2013): all true triples in the KG are filtered out during evaluation, since predicting a low rank for these triples should not be penalised.

4.1 Model analysis

Benefits. HolmE has three major benefits: (1) The design of HolmE gives clear geometric meaning and relatively good transparency (illustrated in Fig. 2), which enables HolmE to model and infer diverse relation patterns effectively. (2) The utilisation of Riemannian space enhances the capacity of HolmE to represent knowledge graphs even within a relatively lower dimension. (3) The closure under composition empowers HolmE with a robust ability to model composition patterns, particularly in a long-tail scenario where such patterns are less-represented in the training data. In contrast to other models, HolmE excels at effectively capturing and modelling these long-tail patterns, even with limited training instances.

Theorem 1 *The KGE defined by Definition 4 is closed under composition. Proof see Appendix A.*

Theorem 2 *The KGE defined by Definition 4 is capable of modelling and inferring diverse relation patterns, including symmetry/asymmetry, inversion, etc. Proof see Appendix A.*

Complexity Analysis. The time complexity of HolmE mainly comes from scoring calculation and cross entropy loss calculation. While the operation for calculating loss is consistent across various KGE models, the complexity of the score function in d-dimensional HolmE is linear, i.e., $O(d)$ for each triple, which is comparable to that of traditional KGEs such as TransE and is lower than that of neural network-based models such as ConvE, as no additional graph aggregation operations are incorporated.

In terms of space complexity, for a KG with $|\mathcal{E}|$ entities and $|\mathcal{R}|$ relations, HolmE needs $d|\mathcal{E}| + 2d|\mathcal{R}|$ parameters. Compared with other Riemannian KGEs, such as AttH and GIE (both need at least $(d+2)|\mathcal{E}| + (3d+1)|\mathcal{R}|$ parameters), HolmE needs fewer parameters. Compared with other Euclidean KGE methods, HolmE achieves similar representation performance in a lower embedding dimension (Chami et al. 2020; Balažević et al. 2019).

Limitations. HolmE is limited in providing theoretically strict mappings that satisfy the relation pattern of *transitivity* and mapping patterns of *one-to-many* (1-to-N), *many-to-one* (N-to-1), and *many-to-many* (N-to-N). However, empirical studies show that HolmE still has good performance on these relation patterns or mapping patterns (Appendix F), due to the mechanism in KGE that selects the best scored predicted tail embedding instead of the exact tail embedding.

Comparison to TransE and RotatE. HolmE has adopted similar components of translation and rotation as in other translational KGE methods. HolmE is a general form that unites a group of KGE methods supporting strong composition, including

TransE and RotatE. KGE methods like neural networks, bilinear models, and mixture models cannot guarantee to be closed under composition because they lack explicit geometric explanation and the notion of *relation embedding as mapping* is not clearly defined. In particular, we have the following claim:

Claim 1 *TransE*, *RotatE* (*Feng et al*. 2016; *Sun et al*. 2019) *are special cases of HolmE*. *Proof see Appendix B*.

Comparison to hyperbolic KGE. MurP (Balažević et al. 2019) adopts translation on high dimensional hyperbolic space and does not adopt rotation. AttH (Chami et al. 2020) adopts rotation in product space, but translation and distance function on high-dimensional manifolds, while GIE (Cao et al. 2022) is a mixture model of different curvature spaces and not pure hyperbolic. In contrast, HolmE has rotation, translation and distance on product space and is closed under composition, thus can represent long-tail composition patterns better.

Claim 2 *MurP* (*Balažević et al*. 2019), *AttH* (*Chami et al*. 2020) *are un-closed under composition*. *Proof see Appendix C*.

5 Experiments

This section evaluates the following claim and hypotheses: (C0) The composition pattern holds significant importance in KG data in the sense that almost all relations are associated with some composition patterns. (H1) HolmE demonstrates superior performance in representing composition patterns. (H2) HolmE excels in representing long-tail composition patterns that are less-represented in the training data. (H3) HolmE performs better in extrapolating unseen relations, given the associated composition pattern. (H4) In benchmark KGs, where other relation patterns also exist, HolmE also exhibits better or comparable performance.

5.1 Composition patterns are significant in KGs (C0)

We assess the presence of composition patterns within three benchmark KGs: WN18RR (Dettmers et al. 2018), FB15k-237 (Toutanova and Chen 2015), YAGO3-10 (Mahdisoltani et al. 2014). We follow the standard data augmentation protocol by adding inverse relations (Lacroix et al. 2018) to the datasets (Table 1).

The composition patterns in the datasets are constrained to have a representation ratio $\mathbf{r}_{CP}$ greater than 3‰, the corresponding thresholds for triple counts (lim_n) are given in Table 1. As the dataset size increases, lim_n also grows, which aligns with the intuition that datasets with greater training data pose a greater challenge for KGE models to uncover underlying patterns. Thus, KGEs need more triples to learn the patterns successfully. These thresholds are determined by assessing the validity of patterns through random sampling, confirming those above the threshold as

Table 1 Composition patterns (*CP*s) in KGs

Data	#ent	#rel	#tr	#tst	#*CP*	lim_n	$\#\text{rel}_{CP}$	$\#\text{tri}_{rel}$	tst_{CP} (%)
WN18RR	41k	11	87k	3k	165	26	10	86k	42.1
FB15k-237	15k	237	272k	20k	4k	81	211	268k	51.7
YAGO3-10	123k	37	1M	5k	357	321	30	1M	8.7

In every KG, there are plenty *CP*s, involving a majority of relations. lim_n: threshold of the triple count for a valid *CP* ($\mathbf{n}_{CP}$); $\#\text{rel}_{CP}$: number of relations that are involved in *CP*s; $\#\text{tri}_{rel}$: number of triples with compositional relations; tst_{CP}: the ratio of test triples that can be derived by *CP*s

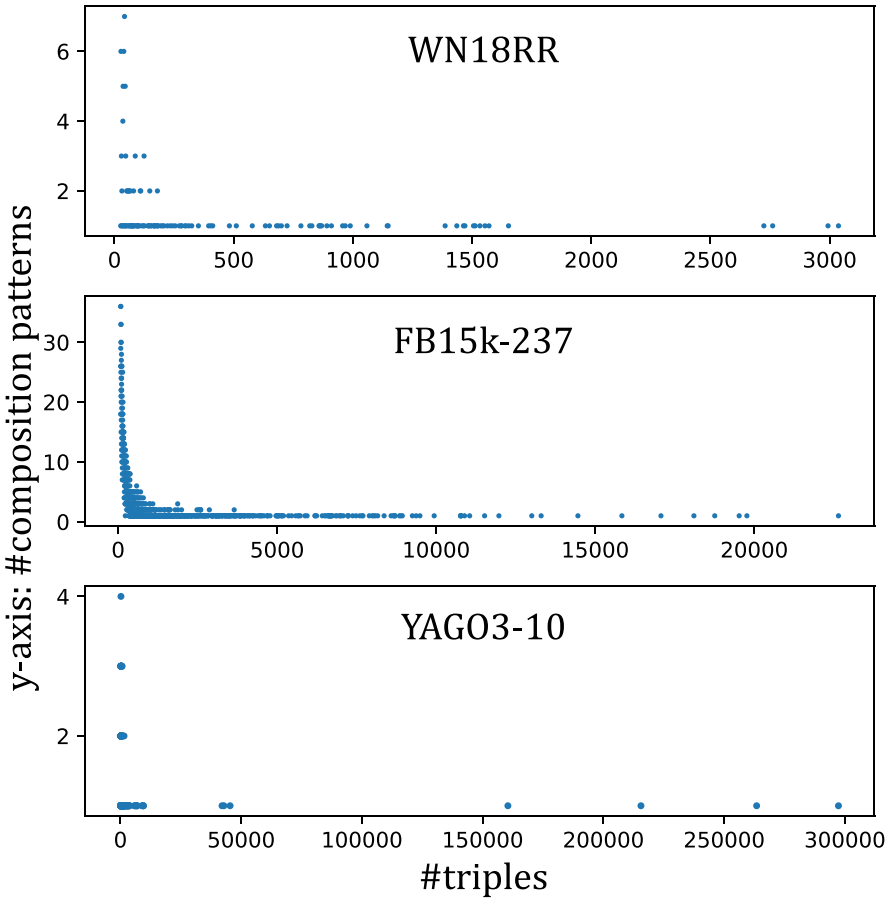

Fig. 3 In the three benchmark datasets, most composition patterns are less-represented (with a smaller triple count $\mathbf{n}_{CP}$) in the training data

true and finding a substantial number of patterns below the threshold to be false. In particular, instances occurring below the specified thresholds are considered as coincidences rather than composition patterns, which ensures the validity of the

identified patterns (see Sect. 3.4). Following this criterion for composition patterns, we find there are plenty of composition patterns (#*CP*) in all the datasets, which involve more than 95% of relations (comparing $\#\text{rel}_{CP}$ and #rel), meaning they either compose or be composed by other relations. Furthermore, it is noteworthy that these relations encompass nearly all the triples present in the training data (comparing $\#\text{tri}_{rel}$ and #tr). These numbers robustly support claim C0, indicating that composition patterns hold significant importance for KGs. In WN18RR and FB15k-237, about half of the test triples can be inferred through the identified composition patterns, while the test set of YAGO3-10 does not specifically emphasise the composition patterns. Additionally, we conducted an analysis to evaluate the representation of these composition patterns within the training sets of benchmark datasets, considering the triple count $\mathbf{n}_{CP}$ (Fig. 3). It is evident from our findings that a significant portion of the composition patterns are underrepresented or observed only in a limited number of instances.

5.2 HolmE excels in modelling composition patterns (H1), especially the long-tail composition patterns (H2)

We provide empirical evidence that HolmE, as a KGE closed under composition, possesses superior capabilities in representing composition patterns, especially those long-tail patterns. We employ the datasets Countries (Nickel et al. 2016) to evaluate HolmE's ability in representing composition patterns. In addition, to verify the capacity of HolmE in modelling long-tail composition patterns, we provide synthetic data and devise subsets of FB15k-237 for better control over the representing ratio of the composition patterns.

5.2.1 Experiments on countries

Datasets. Countries are benchmark KG datasets with readily available training and test data. They are designed specifically for the study of KGE on composition patterns. They contain 2 relations (*locatedIn* and *neighbor*) and 272 entities about countries and regions. The Countries datasets include 3 subsets, S1, S2 and S3, in increasing order of difficulty. S1–S3 use the same test triples, but the number of training triples decreases. In particular, when answering the test triples, S1 has many composition patterns in the training data, while S2 only has few. In S3, only complex long-chain patterns are available. Past works simplified the link prediction task on Countries by constraining the predicted entities to be only continent names and using AUC-PR as the performance metric. This work uses standard metrics and removes the constraints to evaluate HolmE in modelling composition patterns on Countries.

Experiment settings. We adopt the standard filtered setting (Rossi et al. 2021). The optimal hyper-parameters are selected through a grid search with the ranges as follows: embedding dimension $d \in \{32, 100, 200\}$, batch size $b \in \{500, 1000, 2000\}$, learning rate $lr \in \{5\text{e-}3, 1\text{e-}2, 2\text{e-}2\}$, negative sampling size $neg \in \{50, 100, 200\}$. We use Adam (Kingma and Ba 2014) as the optimizer. All experiments

Table 2 HolmE outperforms other KGEs in modelling composition patterns on Countries

Model	Countries-S1			Countries-S2			Countries-S3		
	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10
ComplEx	.284 .031	.154 .034	.525 .033	.296 .035	.172 .043	.557 .037	.209 .015	.083 .023	.514 .019
MurP	.744 .017	.604 .017	.958 .034	.528 .015	.326 .041	.750 .030	.213 .028	.104 .028	.493 .026
AttH	.731 .008	.549 .043	.993 .010	.543 .011	.312 .017	.935 .014	.213 .010	.104 .003	.486 .010
GIE	.751 .024	.639 .036	.972 .010	.598 .046	.431 .029	.840 .037	.221 .014	**.167** .042	.534 .029
TransE	.708 .007	.479 .002	**1.00** .000	.486 .020	.097 .020	**1.00** .000	.132 .007	.001 .002	.472 .010
RotatE	.752 .012	.538 .003	**1.00** .000	.518 .018	.360 .023	.968 .017	.207 .009	.112 .012	.534 .012
HolmE	**.827** .014	**.695** .023	**1.00** .000	**.658** .022	**.480** .014	.939 .023	**.245** .005	.139 .016	**.535** .020

Bold values indicate the best and underlined values indicate the second best

All experiments are conducted five times, with the mean and standard deviation (in smaller font size) reported

are conducted on NVIDIA RTX A3000. The metrics include mean reciprocal rank (MRR) and hits at 1/10 (H@1/10).

Baselines. We include TransE, RotatE, ComplEx-N3 (Lacroix et al. 2018), MurP (Balažević et al. 2019), AttH (Chami et al. 2020) and GIE (Li et al. 2022).

TransE and RotatE are closed under composition (in short KGE-closed), while the others are not (in short KGE-unclosed). ComplEx-N3 is a bilinear model with state-of-the-art comparable performance. In the following discussion ComplEx-N3 is referred as ComplEx, with the suffix omitted for brevity. MurP, AttH and GIE are recent geometric models in hyperbolic space with good performance.

Results and discussion. The results on S1 (Table 2) show that HolmE has superior performance on modelling composition patterns. S2 and S3 present increased challenges as established composition patterns diminish and longer-chain patterns are required, resulting in decreased performance across all models. Despite this, HolmE maintains its superiority, albeit with a narrower margin over other models.

5.2.2 Experiments on synthetic data and subsets of FB15k-237

Datasets. We evaluate on 5 datasets G0-G4 (Table 3).[1] G0 is a synthetic dataset with only three relations, the triples within G0 can be grouped into various combinations exemplified by the form ((a, r1, b), (b, r2, c), (a, r3, c)), all of which manifest an identical composition pattern $r1 \circ r2 = r3$. To introduce additional complexity,

[1] G4 combines the G1–G3, thus containing 9 relations and 3 composition patterns.

certain triples may appear in multiple combinations and peer triples[2] exist. G1–G4 represent subsets of FB15k-237 and are named according to their entity and relation count. Each of G1–G3 comprises three relations and a distinct composition pattern, which are similar to G0. Specifically, the training sets of G0-G3 consist of all triples with r1 and r2, and a portion of triples with r3, which results in the composition pattern being less (G1), medium (G2), and well-represented (G3) in the training data. The remaining triples with r3 form the test and valid set for each dataset. Each valid or test triple can be derived from the training sets through composition patterns. G4 is a merged dataset from G1–G3, incorporating multiple composition patterns and all nine relations in G1–G3. G0 has a simple graph structure with fewer peers, whereas other data, as subsets of FB15k-237, exhibit more complex graph structures due to the presence of a substantial number of peers. In particular, G4 is the most complex due to the presence of multiple composition patterns. All these datasets are open-sourced at https://github.com/nsai-uio/HolmE-KGE.

Results and discussion. The results in Table 4 reveal the following observations:

- Given its synthetic nature, G0 exhibits a relatively simple graph structure, with the composition pattern being less represented. In this context, all KGE-closed (HolmE, TransE, and RotatE) excel, showcasing exceptional performance by accurately predicting nearly all test triples, as reflected in an MRR greater than 0.95.
- Being a subset of FB15k-237, G1 possesses a more complex graph structure compared to G0, resulting in performance degradation across most models. However, all KGE-closed exhibit superior performance compared to others, with HolmE demonstrating the highest level of effectiveness, boasting a notably superior MRR performance, approximately 10% better than its counterparts. Notably, despite their simplicity, TransE and RotatE outperform many other advanced models on G0 and G1. This provides strong empirical evidence for the advantages of KGE-closed in effectively representing less-represented composition patterns.
- HolmE also demonstrates prominent advantages in G2–G4 by about 2% in MRR, which also confirms H2, indicating HolmE is superior in representing composition patterns regardless of their presence in the training data. In addition, we see that the other KGE-closed, TransE and RotatE are no better than other advanced models. The reason is that other factors of KGE, such as model expressivity, becomes increasingly important when the composition patterns are better represented.

Model performance analysis on data with varying $\mathbf{r}_{CP}$.

To further analyse the influence of representing ratio ($\mathbf{r}_{CP}$) on model performance, we conduct detailed analysis on G1-G3 by varying $\mathbf{r}_{CP}$, which indicates the percentage of triples forming composition patterns in the training set (see

[2] Valid alternatives for the head and tail entities (Rossi et al. 2021). For example: (a, r, b) and (a, r, c) are peer triples.

Table 3 Synthetic datasets encompass diverse composition pattern representations: G0–G3 feature the pattern r1 ∘ r2 = r3, where r3 is represented at different frequencies, resulting in composition patterns ranging from less-represented to well-represented in training data

Data	#ent	#train	Composition pattern in train				#test/valid
			#r1	#r2	#r3	$\mathbf{r}_{CP}$	
Compos (G0)	2.6k	2k	1k	1k	10	0.5%	495
FB3.3k-3 (G1)	3.3k	7k	1.6k	5.0k	500	6.9%	2.2k
FB2.7k-3 (G2)	2.7k	17.4k	2.3k	9.9k	5.2k	24.2%	1.3k
FB1.6k-3 (G3)	1.6k	7.1k	1k	3.1k	2.9k	36.8%	82
FB4.8k-9 (G4)[1]	4.8k	31.6k	–	–	–	–	3.6k

G4 constitutes an aggregate of G1–G3

Sect. 3.4). For G1–G3, we relocate r3-related triples to vary $\mathbf{r}_{CP}$ and generate variants of G1–G3. In particular, we move r3-related triples from the test or valid set (or vice versa) to the training set, so that the composition patterns are represented to varying degrees in the training set. Figure 4 shows the results on all the G1–G3. We make the following observations:

- As the composition patterns are better represented (indicated by an increasing $\mathbf{r}_{CP}$), KGEs exhibit enhanced capacity in modelling these patterns. KGE-unclosed (ComplEx-N3, AttH and GIE) have a stronger performance increase compared to KGE-closed (TransE, RotatE and HolmE) as $\mathbf{r}_{CP}$ rises, which also means KGEs-unclosed may not be able to effectively model the less represented composition patterns in the training data.
- In general, KGE-closed has an advantage in modelling these less-represented composition patterns, as we can see in some cases TransE and RotatE have better performance than those more advanced models at low $\mathbf{r}_{CP}$ when the composition patterns are less represented in the training data (e.g., the ComplEx-N3 model in all datasets, or the GIE and AttH models in the G1 variant). These results demonstrate the advantages of a KGE model that is closed under composition in effectively representing less-represented composition patterns.
- Across all datasets and all levels of $\mathbf{r}_{CP}$, HolmE consistently demonstrates the highest representation capability, which confirms the advantages of HolmE in representing composition patterns, irrespective of their representation ratio in the training data.

5.3 HolmE is better to extrapolate unseen relations (H3)

By extrapolating unseen relations, we mean that given r3 = r1 ∘ r2, we try to find the embedding g_{r3} of r3 by learning the embeddings g_{r1} and g_{r2} of r1 and r2, respectively.

Table 4 HolmE have better performance on datasets of composition patterns (G0–G4), especially on long-tail composition patterns (G0–G1)

Model	G0, $\mathbf{r}_{CP}$:0.5%			G1, $\mathbf{r}_{CP}$:6.9%			G2, $\mathbf{r}_{CP}$:24.2%			G3, $\mathbf{r}_{CP}$:36.8%			G4		
	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10
ComplEx	.751 .018	.619 .012	.980 .005	.372 .012	.242 .015	.677 .038	.452 .010	.317 .015	.623 .023	.453 .009	.338 .010	.676 .007	.335 .011	.238 .019	.586 .022
MurP	.587 .027	.463 .031	.826 .028	.484 .005	.368 .004	.731 .003	.517 .007	.270 .012	.947 .001	.672 .017	.523 .018	.856 .032	.323 .012	.182 .008	.577 .022
AttH	.512 .016	.411 .029	.731 .035	.412 .039	.309 .034	.623 .032	.466 .019	.210 .043	.919 .020	.712 .015	.542 .023	**.971** .012	.371 .013	.217 .015	.673 .020
GIE	.411 .015	.285 .021	.701 .035	.446 .027	.343 .037	.661 .036	.498 .024	.264 .013	.918 .004	.703 .032	.533 .029	.951 .033	.418 .021	.242 .022	.708 .015
TransE	**.995** .012	**.974** .031	**1.00** .000	.594 .004	.328 .005	**.878** .012	.352 .004	.034 .009	.934 .001	.458 .008	.153 .015	.968 .027	.272 .011	.023 .021	.669 .022
RotatE	.982 .009	.953 .010	**1.00** .000	.585 .005	.387 .007	.778 .012	.378 .010	.191 .009	.761 .007	.474 .010	.356 .012	.938 .015	.281 .005	.175 .007	.554 .010
HolmE	.993 .010	.966 .009	**1.00** .000	**.636** .020	**.521** .012	.854 .017	**.607** .021	**.393** .035	**.967** .007	**.785** .011	**.646** .012	.943 .015	**.525** .009	**.346** .012	**.840** .011

Bold values indicate the best and underlined values indicate the second best

All experiments are conducted five times, with the mean and standard deviation (in smaller font size) reported

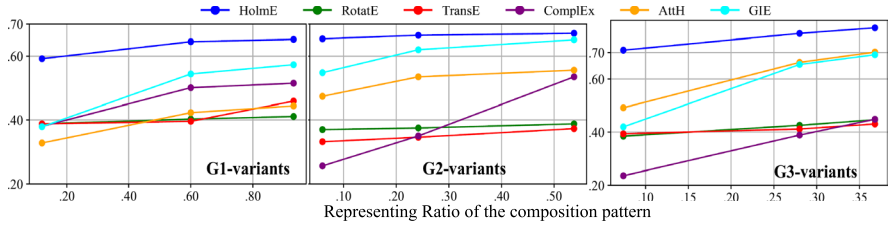

Fig. 4 The performances of all models increase as ratio of composition patterns increase, while HolmE always performs best. Y-axis: MRR

Datasets. The datasets are modified from G1–G4 in Table 3 by excluding all the r3-related triples in the training sets, making r3 unseen for KGEs during training.

Experiment settings. For each dataset, the composition pattern between the relations (r3 = r1 ∘ r2) are provided. During validation and testing, the embeddings of r3 are generated by composing the embeddings of r1 and r2. In particular, in the link prediction task (e1, r3,?), we predict the tail embedding as $g_{r2}(g_{r1}(\mathbf{e}_{e1}))$, then evaluate the performance of KGE using standard link prediction settings.

6 Results and discussion

Across all the datasets, HolmE significantly outperforms all other models by at least 8% in MRR (Table 5), which confirms hypothesis H3, as HolmE demonstrates remarkable performance in extrapolating unseen relations based on composition patterns. Notably, TransE and RotatE, as two other KGEs that are closed under composition, do not achieve optimal performance in this setting when compared to other KGEs. The underlying reason may be that the model expressivity still significantly influences the ability to extrapolate unseen relations.

6.1 HolmE is comparable on benchmark datasets (H4)

We also evaluate HolmE on standard benchmark datasets WN18RR, FB15k-237, YAGO3-10, where various relation patterns exist. Table 6 demonstrates that HolmE exhibits significantly superior performance than other KGE-closed models on three benchmark link prediction datasets. The recent KGE-unclosed models such as GIE and ComplEx can get good performance, but compared to Holme, they need much larger dimensions for embedding. On FB15k-237 and YAGO3-10, HolmE is less than 1% worse than GIE (a less significant performance

Table 5 HolmE outperforms other models significantly in extrapolating unseen relations

Model	G1'			G2'			G3'			G4'		
	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10
ComplEx	.275	.168	.483	.199	.107	.379	.271	.203	.398	.195	.117	.377
MurP	.640	.520	.857	.549	.318	.943	.636	.472	.928	.307	.190	.576
AttH	.472	.376	.676	.529	.297	.928	.586	.410	.919	.279	.161	.543
GIE	.506	.416	.685	.566	.340	.948	.624	.462	.924	.326	.200	.600
TransE	.423	.238	.786	.306	.042	.872	.397	.274	.920	.343	.169	.670
RotatE	.447	.233	.865	.378	.180	.722	.380	.126	.914	.321	.150	.656
HolmE	**.808**	**.702**	**.980**	**.688**	**.581**	**.883**	**.723**	**.571**	**.982**	**.421**	**.294**	**.682**

Bold values indicate the best and underlined values indicate the second best

All datasets are the same as in Table 3, except that r3 are completely excluded from training triples

deterioration, considered in many industrial cases), but have drastically lower dimensions of embeddings. Overall, HolmE has comparable performance and lower dimensions for KGs with composition patterns and other relation patterns.

7 Conclusion and future work

This work proposes HolmE, a general form of KGE, with the property of being closed under composition. We discuss this essential property of KGE models-the closed embedding space under composition-which aligns with real-world relation assumptions, enabling every relation embedding to compose and be composed by other relation embeddings. Such a property proves advantageous for KGEs when modelling long-tail composition patterns that are less represented in the data. Through extensive evaluations, we showcase the significant benefits of HolmE in representing composition patterns, especially those that are less represented in the training data. Furthermore, HolmE also exhibits a remarkable ability to extrapolate to unseen relations based on composition. For general KG datasets with various relation patterns, HolmE demonstrates comparable representation capabilities with a lower embedding dimension. In the future, we will explore broader applications and evaluate the benefits of HolmE as well as the closed embedding space under composition in other areas such as continual learning and inductive learning.

Table 6 HolmE achieves comparable performance on benchmark datasets

Model	WN18RR					FB15k-237					YAGO3-10				
	Dim	MRR	H@1	H@3	H@10	Dim	MRR	H@1	H@3	H@10	Dim	MRR	H@1	H@3	H@10
ComplEx	2000	.480	.435	.495	.572	2000	.357	.264	.392	.547	1000	.569	.498	.609	.701
MurP	200	.481	.440	.495	.566	200	.335	.243	.367	.518		-	-	-	-
AttH	500	.486	.443	.499	.573	500	.348	.252	.384	.540	500	.568	.493	.612	.702
GIE	2000	**.491**	**.452**	.505	.575	800	**.362**	**.271**	**.401**	**.552**	1000	**.579**	**.505**	**.618**	**.709**
TransE	500	.356	.256	.419	.531	500	.336	.243	.369	.524		-	-	-	-
RotatE	2000	.476	.428	.492	.571	2000	.338	.241	.375	.533	2000	.495	.402	.550	.670
HolmE	200	**.491**	.445	**.509**	**.584**	200	.352	.260	.383	.542	200	.570	.497	.612	.701

Bold values indicate the best and underlined values indicate the second best

Results from TransE and ComplEx are from the work of Gregucci et al. (2023) and Chami et al. (2020) respectively, other results are from the original papers. Dim: entity embedding dimensions

Table 7 List of symbols used in this paper

Symbol	Shape	Description
G	–	Knowledge graph (multi-relational graph)
$\mathcal{E}$	–	Set of entities
$\mathcal{R}$	–	Set of relations
$\mathcal{F}$	–	Set of facts (in the form of triples)
h, t	–	Head entity and tail entity
r	–	Relation
$\mathcal{P}$	$\mathcal{P} = \mathbb{H}^{n,\kappa} \times \ldots \times \mathbb{H}^{n,\kappa}$	Product space with m component spaces of dimension n
$\mathcal{M}$	$\mathbb{H}^{n,\kappa}$	Distance space (manifold) of n dimension and curvature κ
$\mathcal{G}$	$\mathbb{H}^{n,\kappa} \to \mathbb{H}^{n,\kappa}$	Mapping space (set of mappings)
s	$\mathcal{P} \times \mathcal{G} \times \mathcal{P} \to \mathbb{R}$	Scoring function
κ	$\mathbb{R}$	Curvature
d	$\mathbb{R}$	Entity embedding size
n	$\mathbb{R}$	Dimension of the component space of $\mathcal{P}$
$\mathbf{e}$	$\mathbb{P}_{m,n,\kappa}$	Entity embedding vector in the product space $\mathcal{P}$
$\mathbf{e}^i$	$\mathbb{H}^{n\times 1,\kappa}$	Entity embedding sub-vector in the component space of $\mathcal{P}$
g	$\mathbb{P}_{m,n,\kappa} \to \mathbb{P}_{m,n,\kappa}$	Relation embedding (mapping) on the product space $\mathcal{P}$
g^i	$\mathbb{H}^{d,\kappa} \to \mathbb{H}^{d,\kappa}$	Relation embedding on the ith component of the product space $\mathcal{M}$
$\mathbf{a}$	$\mathbb{H}^{d\times 1,\kappa}$	Translation vector of g
$\mathbf{A}$	$\mathbb{R}^{d\times d}$	Rotation matrix of g
$\mathbf{R}$	$\mathbb{R}^{n\times n}$	Component rotation matrix of $\mathbf{A}$ in the component space of $\mathcal{P}$
$\oplus^\kappa$ / $\oplus^\kappa_{\mathcal{P}}$	–	Möbius addition in hyperbolic space $\mathcal{M}$ / product space $\mathcal{P}$

Appendix A: Proof of Theorems 1 and 2

See Table 7.

For simplicity, we first prove Theorem 2.

Theorem 2: The KGE defined by Definition 4 is capable of modelling and inferring diverse relation patterns, including symmetry/asymmetry, inversion, etc.

Proof

1. **Symmetric** If $\mathbf{a}_r = \mathbf{0}$ be the original point and $\mathbf{A}_r \cdot \mathbf{A}_r = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix, then we have

$$
\begin{aligned}
& g_r(\mathbf{e}_h) = \mathbf{e}_t \\
\Rightarrow\ & \mathbf{A}_r \cdot \mathbf{e}_h = \mathbf{e}_t \\
\Rightarrow\ & \mathbf{A}_r \cdot \mathbf{A}_r \cdot \mathbf{e}_h = \mathbf{A}_r \cdot \mathbf{e}_t \\
\Rightarrow\ & \mathbf{A}_r \cdot \mathbf{e}_h = \mathbf{e}_t \\
\Rightarrow\ & g_r(\mathbf{e}_h) = \mathbf{e}_t
\end{aligned}
$$

2. **Asymmetric** If $\mathbf{a}_r = \mathbf{0}$ and $\mathbf{A}_r = 2\mathbf{I}$, then $g_r(\mathbf{e}_h) = 2\mathbf{e}_h$. If $g_r(\mathbf{e}_h) = \mathbf{e}_t$ and $g_r(\mathbf{e}_t) = \mathbf{e}_h$, we must have $\mathbf{e}_h = \mathbf{e}_t = \mathbf{0}$.
3. **Inversion** If $\mathbf{a}_r = \mathbf{0}$ and $\mathbf{A}_r$ has inverse matrix $\mathbf{A}_r^{-1}$, then $g_r(\mathbf{e}) = \mathbf{A}_r \cdot \mathbf{e}$ has inverse map $g_{r'}(\mathbf{e}) = \mathbf{A}_r^{-1} \cdot \mathbf{e}$.
4. **Composition** Let $\mathbf{a}_i = \mathbf{0}$ for $i = 1, 2, 3$, and assume that $\mathbf{A}_1 \cdot \mathbf{A}_2 = \mathbf{A}_3$, then we have $g_2(g_1(\mathbf{e}_h)) = \mathbf{A}_1 \cdot \mathbf{A}_2 \cdot \mathbf{e}_h = \mathbf{A}_3 \cdot \mathbf{e}_h = g_3(\mathbf{e}_h), \forall \mathbf{e} \in \mathcal{M}$.

□

Theorem 1: The KGE defined by Definition 4 is closed under composition.
To prove Theorem 1, it is sufficient to prove the following claims.

Claim 3 *The KG embedding defined by Definition 4 is closed under composition.*

Assume:

$$r \otimes \mathbf{x} = \frac{1}{\sqrt{\kappa}} \cdot tanh\left(r \cdot tanh^{-1}(\sqrt{\kappa}||\mathbf{x}||)\right)\frac{\mathbf{x}}{||\mathbf{x}||} \tag{A1}$$

To prove the claim, we need the following results. Consider the *Einstein addition* $\oplus_E$ (Ungar 2012)[Section 8] that satisfies

$$\mathbf{x} \oplus_E \mathbf{y} = 2 \otimes^\kappa \left((\frac{1}{2} \otimes^\kappa \mathbf{x}) \oplus^\kappa (\frac{1}{2} \otimes^\kappa \mathbf{y})\right), \forall \mathbf{x}, \mathbf{y} \in \mathbb{H}^d_\kappa.$$
$$\mathbf{x} \oplus^\kappa \mathbf{y} = \frac{1}{2} \otimes^\kappa \left((2 \otimes^\kappa \mathbf{x}) \oplus_E (2 \otimes^\kappa \mathbf{y})\right), \forall \mathbf{x}, \mathbf{y} \in \mathbb{H}^d_\kappa.$$

We call a function $f : \mathbb{H}^d_\kappa \to \mathbb{H}^d_\kappa$ *preserves Einstein distance* iff $\forall \mathbf{x}, \mathbf{y} \in \mathbb{H}^d_\kappa$, we have

$$||(-f(\mathbf{x})) \oplus_E f(\mathbf{y})|| = ||(-\mathbf{x}) \oplus_E \mathbf{y}||.$$

Then by Ungar (2018, Theorem 3.26), we have:

Theorem 3 *For any function $f : \mathbb{H}^d_\kappa \to \mathbb{H}^d_\kappa$, f preserves Einstein distance iff f is of the form*

$$f(\mathbf{x}) = \mathbf{a} \oplus_E \mathbf{A} \cdot \mathbf{x},$$

for some $\mathbf{a} \in \mathbb{H}^d_\kappa$, $\mathbf{A} \in \mathbf{O}(n)$.

Moreover, we have

Lemma 1 *If $A \in \mathbf{O}(n)$, then we have $r \otimes^\kappa (\mathbf{A} \cdot \mathbf{y}) = \mathbf{A} \cdot (r \otimes^\kappa \mathbf{y}), \forall \mathbf{x} \in \mathbb{H}^d_\kappa$.*

Proof If $\mathbf{A} \in \mathbf{O}(n)$, then $||\mathbf{y}|| = ||\mathbf{A} \cdot \mathbf{y}||$ for all $\mathbf{y} \in \mathbb{H}^d_\kappa$. Therefore, we have

$$
\begin{aligned}
r \otimes^{\kappa} (\mathbf{A} \cdot \mathbf{y}) &= \frac{1}{\sqrt{\kappa}} \cdot tanh\Big(r \cdot tanh^{-1}(\sqrt{\kappa}||\mathbf{Ay}||\Big) \frac{\mathbf{Ay}}{||\mathbf{Ay}||} \\
&= \frac{1}{\sqrt{\kappa}} \cdot tanh\Big(r \cdot tanh^{-1}(\sqrt{\kappa}||\mathbf{y}||\Big) \frac{\mathbf{Ay}}{||\mathbf{y}||} \\
&= \mathbf{A} \cdot (r \otimes^{\kappa} \mathbf{y}).
\end{aligned}
$$

□

Now, we start to prove Theorem 1. For the sake of simplicity, we extend $\circ$ as follows.

$$f_1 \circ \ldots f_{n-1} \circ f_n(\mathbf{x}) = f_1(\ldots f_{n-1}(f_n(\mathbf{x})) \ldots).$$

Proof of Theorem 1 Without loss of generality, we may assume that $\oplus_{\mathcal{P}}^{\kappa} = \oplus$. Let $p : \mathbb{H}_{\kappa}^{d} \to \mathbb{H}_{\kappa}^{d}$ be the mapping defined by $p(u) = 2 \otimes^{\kappa} \mathbf{x}$. Then, the inverse map p^{-1} of p is defined by $p^{-1}(\mathbf{x}) = \frac{1}{2} \otimes^{\kappa} \mathbf{x}$.

Then, for each $g_{(\mathbf{a},\mathbf{A})} \in \mathcal{G}$, where

$$
\begin{aligned}
g_{(\mathbf{a},\mathbf{A})} : \mathbb{H}_{\kappa}^{d} &\to \mathbb{H}_{\kappa}^{d} \\
\mathbf{e} &\mapsto \mathbf{a} \oplus^{\kappa} (\mathbf{A} \cdot \mathbf{e}).
\end{aligned}
$$

We have

$$
\begin{aligned}
g_{(\mathbf{a},\mathbf{A})}(\mathbf{e}) &= \mathbf{a} \oplus^{\kappa} (\mathbf{A} \cdot \mathbf{e}) \\
&= \frac{1}{2} \otimes^{\kappa} \Big((2 \otimes^{\kappa} \mathbf{a}) \oplus_{E} (2 \otimes^{\kappa} (\mathbf{A} \cdot \mathbf{e})) \Big) \\
&= \frac{1}{2} \otimes^{\kappa} \Big((2 \otimes^{\kappa} \mathbf{a}) \oplus_{E} (\mathbf{A} \otimes^{\kappa} (2 \cdot \mathbf{e})) \Big) \quad \text{(by Lemma 1)} \\
&= p^{-1} \Big((2 \otimes^{\kappa} \mathbf{a}) \oplus_{E} (\mathbf{A} \otimes^{\kappa} p(\mathbf{e})) \Big) \\
&= (p^{-1} \circ f_{(\mathbf{a},\mathbf{A})} \circ p)(\mathbf{e}),
\end{aligned}
$$

where

$$
\begin{aligned}
f_{(\mathbf{a},\mathbf{A})} : \mathbb{H}_{\kappa}^{d} &\to \mathbb{H}_{\kappa}^{d} \\
\mathbf{e} &\mapsto (2 \otimes^{\kappa} \mathbf{a}) \oplus_{E} (\mathbf{A} \cdot \mathbf{e}).
\end{aligned}
$$

Then, for any $\mathbf{a}, \mathbf{a}' \in \mathbb{H}_{\kappa}^{d}, \mathbf{A}, \mathbf{A}' \in \mathbf{O}(n)$, we have

$$
\begin{aligned}
g_{(\mathbf{a},\mathbf{A})} \circ g_{(\mathbf{a}',\mathbf{A}')} &= (p^{-1} \circ f_{(\mathbf{a},\mathbf{A})} \circ p) \circ (p^{-1} \circ f_{(\mathbf{a}',\mathbf{A}')} \circ p) \\
&= p^{-1} \circ f_{(\mathbf{a},\mathbf{A})} \circ f_{(\mathbf{a}',\mathbf{A}')} \circ p.
\end{aligned}
$$

By Theorem 3, we know both $f_{(\mathbf{a},\mathbf{A})}$,$f_{(\mathbf{a}',\mathbf{A}')}$ preserve Einstein distance. Then, their composition $f_{(\mathbf{a},\mathbf{A})} \circ f_{(\mathbf{a}',\mathbf{A}')}$ still preserves Einstein distance. By Theorem 3, we can find $\mathbf{b} \in \mathbb{H}^d_\kappa, B \in \mathbf{O}(n)$ such that

$$f_{(\mathbf{a},\mathbf{A})} \circ f_{(\mathbf{a}',\mathbf{A}')}(\mathbf{x}) = \mathbf{b} \oplus_E (\mathbf{B} \cdot \mathbf{x}) = f_{(\frac{1}{2} \otimes^\kappa \mathbf{b},\mathbf{B})}(\mathbf{x}).$$

Notice that, if $\mathbf{A}, \mathbf{A}' \in \mathbf{SO}(n)$, then $f_{(\mathbf{a},\mathbf{A})}$ and $f_{(\mathbf{a}',\mathbf{A}')}$ preserve the *orientation* of the manifold $\mathbb{H}^d_\kappa$, so as their composition $f_{(\mathbf{a},\mathbf{A})} \circ f_{(\mathbf{a}',\mathbf{A}')}$. Therefore, we have $\mathbf{B} \in \mathbf{SO}(n)$. Finally, we have

$$g_{(\mathbf{a},\mathbf{A})} \circ g_{(\mathbf{a}',\mathbf{A}')} = p^{-1} \circ f_{(\frac{1}{2} \otimes^\kappa \mathbf{b},\mathbf{B})} \circ p = g_{(\frac{1}{2} \otimes^\kappa \mathbf{b},\mathbf{B})}.$$

This proves the Theorem 1. □

Appendix B: Proof of Claim 1

Claim 1 The *KG* embedding *TransE* (Bordes et al. 2013), *RotatE* (Sun et al. 2019) are special cases of HolmE.

Proof If we require $\mathbf{A} = \mathbf{I}$ and the curvature $\kappa = 0$, then we have $g_{(\mathbf{a},\mathbf{A})} = \mathbf{a} \oplus^\kappa_{\mathcal{P}} (\mathbf{A} \cdot \mathbf{e}) = \mathbf{a} + \mathbf{e}$. Then HolmE is identical to the *TransE* model.

If we require $\mathbf{a} = \mathbf{0}$, the curvature $\kappa = 0$, and

$$\mathbf{A} = diag[\mathbf{R}^n(\theta_{r,1}), \ldots, \mathbf{R}^n(\theta_{r,d/n})]$$

be the rotation matrix. Then HolmE is identical to *RotatE* model. □

Appendix C: Proof of Claim 2

Claim 2 The *KG* embedding MurP (Balažević et al. 2019), AttH (Chami et al. 2020) are un-closed under composition.

Proof For a given tuple (h, r, t), the score function in MurP of (h, r, t) is defined by:

$$s(h, r, t) = -d^\kappa(\mathbf{R}\mathbf{e}_h, \mathbf{e}_t \oplus^\kappa \mathbf{a}_r),$$

where $\mathbf{R}$ is a diagonal matrix, $\mathbf{e}_h, \mathbf{e}_t, \mathbf{a}_r \in \mathbb{H}^{d,\kappa}$.

Assume $h_{\mathbf{b}}(\mathbf{e}) = \mathbf{e} \oplus^\kappa \mathbf{b}$. In our setting, MurP is equivalent to the model on the embedding space $(\mathcal{M}, \mathcal{G}, s)$ with $\mathcal{M} = \mathbb{H}^{d,\kappa}$, and $\mathcal{G}$ is a group of mappings:

$$g_{(\mathbf{b},\mathbf{R})} : \mathbb{H}^{d,\kappa} \to \mathbb{H}^{d,\kappa} \tag{C2}$$

$$\mathbf{e} \mapsto h_{\mathbf{b}}^{-1}(\mathbf{R} \cdot \mathbf{e}). \tag{C3}$$

Proof by counter-example. We show MurP is un-closed under composition by a counter-example. Next, assume that $d = 2, \kappa = 1$. Let $\mathbf{b}_1 = (1, 0), \mathbf{R}_1 = \mathbf{I}$ and $\mathbf{b}_2 = (0, 0), \mathbf{R}_2 = 2 \cdot \mathbf{I}$. We show that there do not exists $\mathbf{b}_3, \mathbf{R}_3$ such that $g_{(\mathbf{b}_3,\mathbf{R}_3)} = g_{(\mathbf{b}_2,\mathbf{R}_2)} \circ g_{(\mathbf{b}_1,\mathbf{R}_1)}$.

If not, assume $\mathbf{b}_3, \mathbf{R}_3$ satisfies $g_{(\mathbf{b}_3,\mathbf{R}_3)} = g_{(\mathbf{b}_2,\mathbf{R}_2)} \circ g_{(\mathbf{b}_1,\mathbf{R}_1)}$. Then, for any $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3 \in \mathbb{H}^{d,\kappa}$, we have:

$$\mathbf{e}_2 = g_{(\mathbf{b}_1,\mathbf{R}_1)}(\mathbf{e}_1),\ \mathbf{e}_3 = g_{(\mathbf{b}_2,\mathbf{R}_2)}(\mathbf{e}_2) \ \Rightarrow\ \mathbf{e}_3 = g_{(\mathbf{b}_3,\mathbf{R}_3)}(\mathbf{e}_1).$$

That is (recall that $\mathbf{b}_1 = (1, 0), \mathbf{R}_1 = \mathbf{I}$ and $\mathbf{b}_2 = (0, 0), \mathbf{R}_2 = 2 \cdot \mathbf{I}$):

$$\mathbf{e}_2 \oplus^{\kappa} \mathbf{b}_1 = \mathbf{e}_1,\ \ \mathbf{e}_3 = 2\mathbf{e}_2 \ \Rightarrow\ \mathbf{e}_3 \oplus^{\kappa} \mathbf{b}_3 = \mathbf{R}_3\mathbf{e}_1.$$

Then we have

$$(0.5\mathbf{e}_3) \oplus^{\kappa} \mathbf{b}_1 = \mathbf{e}_1 \ \Rightarrow\ \mathbf{e}_3 \oplus^{\kappa} \mathbf{b}_3 = \mathbf{R}_3\mathbf{e}_1.$$

Therefore

$$\mathbf{e}_3 \oplus^{\kappa} \mathbf{b}_3 = \mathbf{R}_3\Big((0.5\mathbf{e}_3) \oplus^{\kappa} \mathbf{b}_1\Big), \forall \mathbf{e}_3 \in \mathbb{H}^{d,\kappa}.$$

Let $\mathbf{e}_3 = (0, 0)$, we have $\mathbf{b}_3 = \mathbf{R}_3\mathbf{b}_1$. Assume $e_3 = (x, y), \mathbf{R}_3 = diag(a, b)$, then $\mathbf{b}_3 = (a, 0)$. We have:

$$(x, y) \oplus^{\kappa} (a, 0) = diag(a, b) \cdot \Big((0.5x, 0.5y) \oplus^{\kappa} (1, 0)\Big).$$

Then, by comparing the first coordinate of each side, we have:

$$\begin{aligned} &\frac{(1 - 2ax - a^2)x + a(1 + x^2 + y^2)}{1 - 2ax + a^2(x^2 + y^2)} \\ &= a\frac{1 + \frac{1}{4}(-x^2 + y^2)}{1 - x + \frac{1}{4}(x^2 + y^2)}, \quad \forall x, y \in \mathbb{R}. \end{aligned} \tag{C4}$$

Let $x = 0$, we have

$$\frac{a(1 + y^2)}{1 + a^2y^2} = a, \forall y \in \mathbb{R}.$$

Then $a = 0$, then by Eq. (C4), we have $x = 0, \forall x \in \mathbb{R}$. Contradiction.

Now, we prove AttH (Chami et al. 2020) is un-closed under composition in the same manner. For simplicity, we assume that the embedding manifold is of dimension 2. Then the scoring function of AttH is defined by:

$$s(h, r, t) = -d^{\kappa}(Q(h, r) \oplus^{\kappa} \mathbf{a}_r, \mathbf{e}_t),$$

when $\kappa = 0$, $Q(h, r) = (\lambda Rot(\theta) + (1 - \lambda)Ref(\phi)) \cdot \mathbf{e}_h$, where λ, θ, ϕ are parameters determined by r s.t. $\lambda \in (0, 1), \theta, \phi \in \mathbb{R}$ and

$$Rot(\theta) := \begin{bmatrix} cos(\theta) & -sin(\theta) \\ sin(\theta) & cos(\theta) \end{bmatrix} \tag{C5}$$

$$Ref(\phi) := \begin{bmatrix} cos(\phi) & sin(\phi) \\ sin(\phi) & -cos(\phi) \end{bmatrix} \tag{C6}$$

Next, assume $\kappa = 0$. In this case, the Riemannian manifold is identical to the linear space, Let $\mathbf{a}_r = \mathbf{a}_s = \mathbf{0}$, $\lambda \neq \lambda', \lambda, \lambda' \in (0, 1)$, and

$$\begin{aligned} Q(h, r) =& (\lambda Rot(0) + (1 - \lambda)Ref(0)) \cdot \mathbf{e}_h, \\ Q(h, s) =& (\lambda' Rot(0) + (1 - \lambda')Ref(\pi)) \cdot \mathbf{e}_h. \end{aligned}$$

We show that there does not exists λ'', $Q(h.t)$ such that

$$Q(Q(h, r), s) = Q(h, t) \oplus^{\kappa} \mathbf{a}_r. \tag{C7}$$

Proof by Counterexample. Otherwise, assume Eq. (C7) is true for

$$Q(h, s) = (\lambda'' Rot(\theta) + (1 - \lambda'')Ref(\phi)) \cdot \mathbf{e}_h.$$

Then by taking $\mathbf{e}_h = (0, 0)$, we have $\mathbf{a}_r = (0, 0)$. Therefore, we have

$$\begin{aligned} &(\lambda' Rot(0) + (1 - \lambda')Ref(\pi)) \cdot (\lambda Rot(0) + (1 - \lambda)Ref(0)) \cdot \mathbf{e}_h \\ &= (\lambda'' Rot(\theta) + (1 - \lambda'')Ref(\phi)) \cdot \mathbf{e}_h. \end{aligned}$$

That is:

$$\mathbf{R}_0 \cdot \mathbf{e}_h = \mathbf{R}(\theta, \phi) \cdot \mathbf{e}_h, \quad \forall \mathbf{e}_h \in \mathbb{R}^2, \tag{C8}$$

where

$$\mathbf{R}_0 := \begin{bmatrix} 1 - 2\lambda' & 0 \\ 0 & 1 - 2\lambda \end{bmatrix}, \quad \lambda \neq \lambda', \lambda, \lambda' \in (0, 1), \text{ and} \tag{C9}$$

$$\begin{aligned} \mathbf{R}(\theta, \phi) := \lambda'' &\begin{bmatrix} cos(\theta) & -sin(\theta) \\ sin(\theta) & cos(\theta) \end{bmatrix} \\ &+ (1 - \lambda'') \begin{bmatrix} cos(\phi) & sin(\phi) \\ sin(\phi) & -cos(\phi) \end{bmatrix} \end{aligned} \tag{C10}$$

Then $\mathbf{R}_0 = \mathbf{R}(\theta, \phi)$, and thus $\lambda'' \cdot (-sin(\theta)) = -(1 - \lambda'')sin(\phi) = \lambda'' \cdot (-sin(\theta))$. Therefore, they are all 0. There are two cases:

1. $\lambda'' \in \{0, 1\}$, then $\mathbf{R}(\theta, \phi) = Ref(\phi)$ or $Rot(\theta)$. For any θ, ϕ, none of them equals $\mathbf{R}_0$
2. $sin(\theta) = sin(\phi) = 0$, then $\{cos(\theta), cos(\phi)\} \subseteq \{1, -1\}$. However, in any cases, we have $|\mathbf{R}(\theta, \phi)_{ii}| = 1$ for some $i \in \{1, 2\}$. Then, $\mathbf{R}(\theta, \phi) \neq \mathbf{R}_0$ as $0 < (\mathbf{R}_0)_{11}, (\mathbf{R}_0)_{22} < 1$.

Finally, in both cases, $\mathbf{R}(\theta, \phi) \neq \mathbf{R}_0$. This violated Eq. (). □

Appendix D: Problem setup details

D.1 Riemannian Geometry

Here we provide a short introduction and refer the readers to textbooks such as Willmore (2013).

Manifold. A manifold is a generalisation of the concept of a distance space. It is a mathematical space that locally resembles Euclidean space (the ordinary space of our everyday experience), but can have a more complex global structure.

Curvature. The curvature of the space affects how distances and angles are measured. It is a measure of how much the manifold deviates from being flat (Euclidean space). Curvature can be positive (like a sphere), negative (like a saddle), or zero (Euclidean space). In a positively curved space, the distance between two points is shorter than it would be in a flat space. In a negatively curved space, the distance is longer. For example, on the surface of a sphere (a 2-dimensional positively curved manifold), the shortest distance between two points (a geodesic) is an arc of a great circle, which is shorter than the corresponding straight line distance in a flat plane.

Riemannian manifold. A Riemannian manifold is a distance space where the distance between two points are characterised by a Riemannian metric. In a Riemannian manifold, the Riemannian metric is influenced by the curvature of the space, which can vary from point to point. In our paper, HolmE is an embedding method on *hyperbolic space*, which is a type of Riemannian manifold with constant negative curvature $-\kappa$ and dimension d as $\mathbb{H}^{d,\kappa}$.

Tangent space. It is common to establish mappings between Riemannian manifold and its tangent spaces. A tangent space is a vector space that locally approximates the manifold at a specific point as if the manifold were a Euclidean space. At each point on a Riemannian manifold $\mathbf{x} \in \mathbb{H}^{d,\kappa}$, there is a tangent space $\mathcal{T}^{\kappa}_{\mathbf{x}}$, which is a d-dimensional vector space consisting of all possible tangent vectors at that point $\mathbf{x}$. The tangent space has the same dimension d as the Riemannian manifold.

Exponential map and logarithmic map. The tangent space $\mathcal{T}^{\kappa}_{x}$ can be mapped to the Riemannian manifold $\mathbb{H}^{d,\kappa}$ via the exponential map, and conversely, the logarithmic map maps the Riemannian manifold $\mathbb{H}^{d,\kappa}$ to the tangent space $\mathcal{T}^{\kappa}_{\mathbf{x}}$. In particular, we have closed-form expressions for these maps at the origin:

$$exp_0^\kappa(\mathbf{x}) = tanh(\sqrt{\kappa}||\mathbf{x}||)\frac{\mathbf{x}}{\sqrt{\kappa}||\mathbf{x}||}$$
$$log_0^\kappa(\mathbf{y}) = arctanh(\sqrt{\kappa}||\mathbf{y}||)\frac{\mathbf{y}}{\sqrt{\kappa}||\mathbf{y}||}$$

Appendix E: Method details

HolmE is a general form of Riemannian KGE, in the sense that (1) the rotation can be 2 dimensional and higher dimensional; (2) the curvature can be 0 (then the entity embedding space becomes Euclidean space), or a positive constant (then the entity embedding space becomes sphere space). In the main text, we exemplified HolmE with 2-dimensional rotation for simplicity and computational efficiency. In principle, HolmE can also have higher dimensional rotations. For example, 3-dimensional rotation matrix $\mathbf{R}_r^3$ can be defined as follows:

$$\mathbf{A}_x(\phi_r) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\phi_r) & -sin(\phi_r) \\ 0 & sin(\phi_r) & cos(\phi_r) \end{bmatrix} \quad \mathbf{A}_Y(\theta_r) = \begin{bmatrix} cos(\theta_r) & 0 & sin(\theta_r) \\ 0 & 1 & 0 \\ -sin(\theta_r) & 0 & cos(\theta_r) \end{bmatrix}$$
$$\mathbf{A}_Z(\psi_r) = \begin{bmatrix} cos(\psi_r) & -sin(\psi_r) & 0 \\ sin(\psi_r) & cos(\psi_r) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \mathbf{R}_r^3 = \mathbf{A}_x(\phi_r) \cdot \mathbf{A}_Y(\theta_r) \cdot \mathbf{A}_Z(\psi_r) \tag{E11}$$

Appendix F: Experiments

HolmE is comparable in representing other relation patterns. We trained AttH, MurP, and HolmE with 32-dimensional embedding spaces on the FB15k-237 dataset. The test set was partitioned into different groups with overlaps, with each group derived from the training data based on specific relation patterns or relation mapping properties. We evaluated the trained models in each group to assess their modelling capabilities within these relation patterns. The results of these evaluations are reported in Tables 8 and 9.

Table 8 The modelling performance in MRR for relation patterns on FB15k-237

	AttH	MurP	HolmE
Sym. 0.322	.322	.330	
Asym. 0.326	.327	.338	
Inv	0.319	.321	.331
Tran	0.314	.311	.317
Comp. 0.279	.282	.292	

Sym., Symmetry; Asym., Asymmetry; Inv., Inversion; Tran., Transitivity; Comp., Composition

Table 9 The modelling performance in MRR for relation mapping properties in low-dimensional space on FB15k-237

Task	RMPs	AttH	MurP	HolmE
Predi-	1-to-1	.460	.452	.485
cting	1-to-N	.430	.424	.456
Head	N-to-1	.079	.085	.099
(MRR)	N-to-N	.243	.237	.253
Predi-	1-to-1	.452	.443	.475
cting	1-to-N	.060	.065	.077
Tail	N-to-1	.728	.704	.743
(MRR)	N-to-N	.350	.347	.360

Author Contributions Zhuoxun Zheng and Baifan Zhou wrote the main manuscript text. Zhuoxun Zheng and Zhipeng Tan made the experiments. Hui Yang conducted theoretical derivation and proof. Chunnong Li prepared figures 1-2. All authors reviewed the manuscript.

Funding Open access funding provided by University of Oslo (incl Oslo University Hospital)

Availability of data and materials No datasets were generated or analysed during the current study.

Declarations

Competing interests The authors declare no competing interests.

References

Abboud R, Ceylan I, Lukasiewicz T, Salvatori T (2020) Boxe: a box embedding model for knowledge base completion. In: NeurIPS, vol 33, pp 9649–9661

Arakelyan E, Daza D, Minervini P, Cochez M (2020) Complex query answering with neural link predictors. In: ICLR

Bai Y, Ying Z, Ren H, Leskovec J (2021) Modeling heterogeneous hierarchies with relation-specific hyperbolic cones. In: NeurIPS, vol 34, pp 12316–12327

Balažević I, Allen C, Hospedales T (2019) Multi-relational poincaré graph embeddings. In: NeurIPS, vol 32

Bordes A, Usunier N, Garcia-Duran A, Weston J, Yakhnenko O (2013) Translating embeddings for modeling multi-relational data. In: NeurIPS, vol 26

Cao Z, Xu Q, Yang Z, Cao X, Huang Q (2022) Geometry interaction knowledge graph embeddings. In: AAAI

Chami I, Wolf A, Juan D-C, Sala F, Ravi S, Ré C (2020) Low-dimensional hyperbolic knowledge graph embeddings. In: ACL, pp 6901–6914

Chami I, Ying Z, Ré C, Leskovec J (2019) Hyperbolic graph convolutional neural networks. In: NeurIPS, vol 32
Chen M, Zhang W, Zhu Y, Zhou H, Yuan Z, Xu C, Chen H (2022) Meta-knowledge transfer for inductive knowledge graph embedding. In: SIGIR, pp 927–937
Datta A, Derek A, Mitchell JC, Roy A (2007) Protocol composition logic (PCL). Electron Notes Theor Comput Sci 172:311–358
Dettmers T, Minervini P, Stenetorp P, Riedel S (2018) Convolutional 2D knowledge graph embeddings. In: AAAI, vol 32
Feng J, Huang M, Wang M, Zhou M, Hao Y, Zhu X (2016) Knowledge graph embedding by flexible translation. In: KR
Ganea O, Bécigneul G, Hofmann T (2018) Hyperbolic neural networks. In: NeurIPS, vol 31
Gao C, Sun C, Shan L, Lin L, Wang M (2020) Rotate3d: representing relations as rotations in three-dimensional space for knowledge graph embedding. In: CIKM, pp 385–394
Gregucci C, Nayyeri M, Hernández D, Staab S (2023) Link prediction with attention applied on multiple knowledge graph embedding models. In: WWW, pp 2600–2610
Ji G, He S, Xu L, Liu K, Zhao J (2015) Knowledge graph embedding via dynamic mapping matrix. In: ACL and IJCNLP, vol 1, Long Papers, pp 687–696
Jin D, Huo C, Liang C, Yang L (2021) Heterogeneous graph neural network via attribute completion. In: WWW, pp 391–400
Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv:1412.6980
Lacroix T, Usunier N, Obozinski G (2018) Canonical tensor decomposition for knowledge base completion. In: ICML, pp 2863–2872. PMLR
Li R, Cao Y, Zhu Q, Bi G, Fang F, Liu Y, Li Q (2022) How does knowledge graph embedding extrapolate to unseen data: a semantic evidence view. In: AAAI, vol 36, pp 5781–5791
Lin Y, Liu Z, Sun M, Liu Y, Zhu X (2015) Learning entity and relation embeddings for knowledge graph completion. In: AAAI
Mahdisoltani F, Biega, J, Suchanek F (2014) YAGO3: a knowledge base from multilingual Wikipedias. In: CIDR. CIDR Conference
Nathani D, Chauhan J, Sharma C, Kaul M (2019) Learning attention-based embeddings for relation prediction in knowledge graphs. In: ACL
Nickel M, Rosasco L, Poggio T (2016) Holographic embeddings of knowledge graphs. In: AAAI, vol 30
Nickel M, Tresp V, Kriegel H-P (2011) A three-way model for collective learning on multi-relational data. In: ICML
Pavlović A, Sallinger E (2022) Expressive: a spatio-functional embedding for knowledge graph completion. In: ICLR
Rossi A, Barbosa D, Firmani D, Matinata A, Merialdo P (2021) Knowledge graph embedding for link prediction: a comparative analysis. ACM Trans Knowl Discov Data 15(2):1–49
Sala F, De Sa C, Gu A, Ré C (2018) Representation tradeoffs for hyperbolic embeddings. In: ICML, pp 4460–4469. PMLR
Schlichtkrull M, Kipf TN, Bloem P, Berg RVD, Titov I, Welling M (2018) Modeling relational data with graph convolutional networks. In: ESWC, pp 593–607. Springer
Sun Z, Deng Z-H, Nie J-Y, Tang J (2019) RotatE: knowledge graph embedding by relational rotation in complex space. In: ICLR
Teru K, Denis E, Hamilton W (2020) Inductive relation prediction by subgraph reasoning. In: ICML, pp 9448–9457. PMLR
Toutanova K, 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, pp 57–66
Trouillon T, Welbl J, Riedel S, Gaussier É, Bouchard G (2016) Complex embeddings for simple link prediction. In: ICML, pp 2071–2080. PMLR
Ungar A (2018) Beyond pseudo-rotations in pseudo-Euclidean spaces
Ungar AA (2012) Möbius transformation and Einstein velocity addition in the hyperbolic geometry of Bolyai and Lobachevsky, pp 721–770
Ungar AA (2001) Hyperbolic trigonometry and its application in the Poincaré ball model of hyperbolic geometry. Comput Math Appl 41(1–2):135–147
Wang Q, Mao Z, Wang B, Guo L (2017) Knowledge graph embedding: a survey of approaches and applications. IEEE Trans Knowl Data Eng 29(12):2724–2743

Wang P, Han J, Li C, Pan R (2019) Logic attention based neighborhood aggregation for inductive knowledge graph embedding. In: AAAI, vol 33, pp 7152–7159
Wang Z, Zhang J, Feng J, Chen Z (2014) Knowledge graph embedding by translating on hyperplanes. In: AAAI, vol 28
Willmore TJ (2013) An introduction to differential geometry
Xiong W, Yu M, Chang S, Guo X, Wang WY (2018) One-shot relational learning for knowledge graphs. arXiv:1808.09040
Yang B, Yih W-T, He X, Gao J, Deng L (2014) Embedding entities and relations for learning and inference in knowledge bases. arXiv:1412.6575
Yu Z, Jin D, Liu Z, He D, Wang X, Tong H, Han J (2021) As-gcn: adaptive semantic architecture of graph convolutional networks for text-rich networks. In: ICDM, pp 837–846. IEEE
Zhang S, Tay Y, Yao L, Liu Q (2019) Quaternion knowledge graph embeddings. In: NeurIPS, vol 32
Zhang C, Yao H, Huang C, Jiang M, Li Z, Chawla NV (2020) Few-shot knowledge graph completion. In: AAAI, vol 34, pp 3041–3048
Zheng Z, Zhou B, Yang H, Tan Z, Waaler A, Kharlamov E, Soylu A (2024) Low-dimensional hyperbolic knowledge graph embedding for better extrapolation to under-represented data. In: ESWC
Zhu Z, Zhang Z, Xhonneux L-P, Tang J (2021) Neural bellman-ford networks: a general graph neural network framework for link prediction. In: NeurIPS, vol 34, pp 29476–29490