



Cao, J., Fang, J., Meng, Z. and Liang, S. (2024) Knowledge graph embedding: a survey from the perspective of representation spaces. *ACM Computing Surveys*, (doi: [10.1145/3643806](https://doi.org/10.1145/3643806))

There may be differences between this version and the published version. You are advised to consult the publisher's version if you wish to cite from it.

© 2024 Copyright held by the owner/author(s). This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *ACM Computing Surveys* 2024
<https://doi.org/10.1145/3643806>

<https://eprints.gla.ac.uk/307645/>

Deposited on: 06 October 2023

Knowledge Graph Embedding: A Survey from the Perspective of Representation Spaces

JIAHANG CAO*, Sun Yat-sen University, China

JINYUAN FANG*, Sun Yat-sen University, China

ZAIQIAO MENG†, University of Glasgow, UK

SHANGSONG LIANG†, Sun Yat-sen University, China and Mohamed bin Zayed University of Artificial Intelligence, UAE

Knowledge graph embedding (KGE) is an increasingly popular technique that aims to represent entities and relations of knowledge graphs into low-dimensional semantic spaces for a wide spectrum of applications such as link prediction, knowledge reasoning and knowledge completion. In this paper, we provide a systematic review of existing KGE techniques based on representation spaces. Particularly, we build a fine-grained classification to categorise the models based on three mathematical perspectives of the representation spaces: (1) Algebraic perspective, (2) Geometric perspective, and (3) Analytical perspective. We introduce the rigorous definitions of fundamental mathematical spaces before diving into KGE models and their mathematical properties. We further discuss different KGE methods over the three categories, as well as summarise how spatial advantages work over different embedding needs. By collating the experimental results from downstream tasks, we also explore the advantages of mathematical space in different scenarios and the reasons behind them. We further state some promising research directions from a representation space perspective, with which we hope to inspire researchers to design their KGE models as well as their related applications with more consideration of their mathematical space properties.

CCS Concepts: • **Computing methodologies** → **Learning latent representations; Neural networks; Information systems** → **Data mining**.

Additional Key Words and Phrases: Knowledge graphs, representation spaces, embedding techniques, mathematical perspectives,

ACM Reference Format:

Jiahang Cao, Jinyuan Fang, Zaiqiao Meng, and Shangsong Liang. 2023. Knowledge Graph Embedding: A Survey from the Perspective of Representation Spaces. *ACM Comput. Surv.* 1, 1 (October 2023), 42 pages. <https://doi.org/XX.XXXX/XXXXXXXX.XXXXXXX>

1 Introduction

Knowledge Graphs are a type of multi-relational graphs that store factual knowledge in real-world. Nodes in KGs represent real-world entities (e.g., names, events and products) and edges represent the relationships between entities. Normally, a KG can be efficiently stored as knowledge triples, where each triple consists of two entities and one factual relation between them (i.e., *<head entity, relation, tail entity>*). For example, in the triple *<RNA virus, subclass, COVID-19>*,

*Both authors contributed equally to the paper.

†Corresponding authors.

Authors' addresses: Jiahang Cao, Sun Yat-sen University, No.132, EastWaihuan Road, Guangzhou, China, 510006, caojh7@mail2.sysu.edu.cn; Jinyuan Fang, Sun Yat-sen University, No.132, EastWaihuan Road, Guangzhou, China, 510006, fangjy6@gmail.com; Zaiqiao Meng, University of Glasgow, Glasgow, UK, zaiqiao.meng@glasgow.ac.uk; Shangsong Liang, Sun Yat-sen University, No.132, EastWaihuan Road, Guangzhou, China, 510006 and Mohamed bin Zayed University of Artificial Intelligence, Masdar City, Abu Dhabi, UAE, 00001, liangshangsong@gmail.com.

© 2023 Association for Computing Machinery.

This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *ACM Computing Surveys*, <https://doi.org/XX.XXXX/XXXXXXXX.XXXXXXX>.

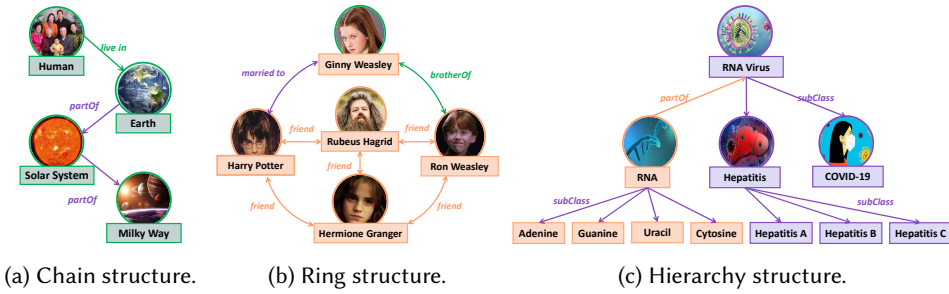


Fig. 1. An illustration of three types of KG structures. (a) shows the most common chain structure in KGs, which can usually be directly modelled in Euclidean space (e.g., TransE, TransH, etc.). (b) is a ring structure of KGs that can be captured in hypersphere space [21]. (c) stands for hierarchical structure which is usually encoded in hyperbolic or spherical spaces [24, 157].

RNA virus and *COVID-19* are real-world entities and *subclass* represents the relation between *RNA virus* and *COVID-19*. Over the recent years, rapid growth has been witnessed in building large-scale KGs, such as YAGO [155], Wikidata [174], Freebase [11] and DBpedia [3]. Due to their effectiveness in storing and representing factual knowledge, they have been successfully applied in question answering [148, 214], recommendation system [156, 239], information retrieval [49, 193] and other domain-specific applications [94, 112]. Despite the KGs are effective in representing structured factual information, they are difficult to manipulate due to the large-scale and complicated graph structure, i.e., the relationships between entities are intricate and complex, such as the ring structure and hierarchy structure depicted in Figure 1c. Therefore, how to effectively and efficiently extract and leverage useful information in large-scale KGs for downstream tasks, such as link prediction [24, 158, 225] and entity classification [75, 202, 217], is a tough task. To tackle this challenging task, the **Knowledge Graph Embedding (KGE)** technique was proposed, and has been receiving a lot of attention in the machine learning community [14, 24, 68, 104, 119, 158]. The essential idea of KGE is to learn to embed entities and relations of a KG into a low-dimensional space (i.e. vectorial embeddings), where the embeddings are required to preserve the semantic meaning and relational structure of the original KG. The learned embeddings of entities and relations can then be leveraged to solve downstream applications, such as KG completion [1, 14, 190, 220, 224], question answering [36, 103, 207, 214, 238], information extraction [42, 67, 105, 194, 234] and entity classification [96, 153].

Many KGE techniques have been proposed to learn the embeddings of entities and relations in KGs [76, 109, 158, 190, 196]. Some KGE methods propose to learn KG embeddings by preserving *relational patterns* between entities in KGs. For example, in order to capture the transformation relationships between entities, TransE [14] was proposed to embed KGs into *Euclidean Space* and represent relations between entities as translation vectors between entity embeddings in the vector space. Moreover, in order to preserve and infer other relational patterns including symmetry, antisymmetry, inversion and composition in KGs, RotatE [158] was proposed to map KGs into *Complex Vector Space*, where relations are represented as rotations between entities.

Another line of KGE methods proposes to learn KG embeddings by preserving *structural patterns* of KGs. This line of works was motivated by the fact that large-scale KGs usually contain many complex and compound structures. For example, in Figure 1, we provide an illustration of three typical types of structure patterns in KGs, namely chain structure, ring structure and hierarchy structure. In order to effectively capture the hierarchy structures in KGs, ATTH [24] was proposed

to embed KGs into *Hyperbolic Space* with trainable curvatures, where richer transformations can be used to separate nodes than Euclidean space [126], while capturing logical patterns simultaneously.

In addition, some KGE methods also try to embed KGs in other mathematical spaces to model some desirable properties in KGs. For example, KG2E [65] is the first “density-based” embedding [173] technique, which uses Gaussian distributions as embeddings instead of deterministic vectors, to model the uncertainties of entities and relations. Moreover, TorusE [38] chooses a compact *Lie Group* as its embedding manifold to deal with the regularisation problems, and ModuLE [23] also introduces *Group* theory to model both entities and relations as group element, which can accommodate and outperform most of the existing KGE models.

From the perspective of representation space, we found that the above KGE methods mostly learn embeddings in different mathematical spaces, e.g., Euclidean space, Hyperbolic space and Probability space, to capture different relational and structural patterns in KGs. Indeed, different mathematical spaces have their unique strengths, which are beneficial to capture different patterns and properties in KGs. Therefore, we argue that representation space plays a significant role in KGE methods, as it determines the patterns and properties of KGs that can be captured and preserved by KG embeddings. In addition to the KGE domain, some studies [18, 123, 134] also demonstrate the importance of mathematical space in traditional machine learning.

Some surveys have been devoted to discussing traditional machine learning models from the perspective of mathematical space [123, 134]. However, there is not yet a systematic review of KGE methods from the perspective of mathematical space. Existing surveys about KGE methods focus either on the encoding model or the applications of KGE methods. For example, Wang et al. [181] classify KGE methods based on their embedding functions and categorise them into three folds: translation-based models, semantic matching models and additional information-based models. Ji et al. [77] provide a full-scaled view to introduce KGE from four aspects: representation learning, scoring function, encoding models and auxiliary information. Lu et al. [114] survey KGE methods with a concentration on utilising textual information.

Accordingly, in this paper we aim at providing a comprehensive survey on representation spaces for knowledge graph embedding techniques, summarising different properties of representation spaces as well as providing guidance for building KGE methods. In order to have a better understanding of KGE methods from a novel spatial view, inspired by the fundamental mathematical space system, we build a systematic, comprehensive and multi-angle categorisation to classify existing KGE methods based on their representation spaces. Specifically, we propose to classify existing KGE methods into three categories, namely **Algebraic Structure**, **Geometric Structure** and **Analytical Structure**. These three structures have their own mathematical focus, but they are intrinsically linked and jointly make the mathematical system more complete and concrete [32, 138]. Figure 2 provides an overview of our classification framework and some representative KGE methods that belong to each category (the detailed version can be found in Section 3). In this survey, we will introduce the definitions and properties of the above three mathematical structures and introduce some representative KGE methods that belong to these categories in detail. Moreover, we will summarise the experimental results of different KGE methods and provide some suggestions and guidance for building more expressive and powerful KGE methods. Furthermore, we will point out new trends and further directions of KGE methods from the perspective of representation space.

Recently, LLMs such as T5 [140] and GPT-4¹ have achieved remarkable success in a variety of natural language processing tasks, such as text generation [93], machine translation [91] and question answering [74]. Despite their success in many applications, LLMs still suffer from the following shortcomings: (1) They are known to suffer from the hallucination issue [7, 78], i.e.,

¹<https://openai.com/gpt-4>

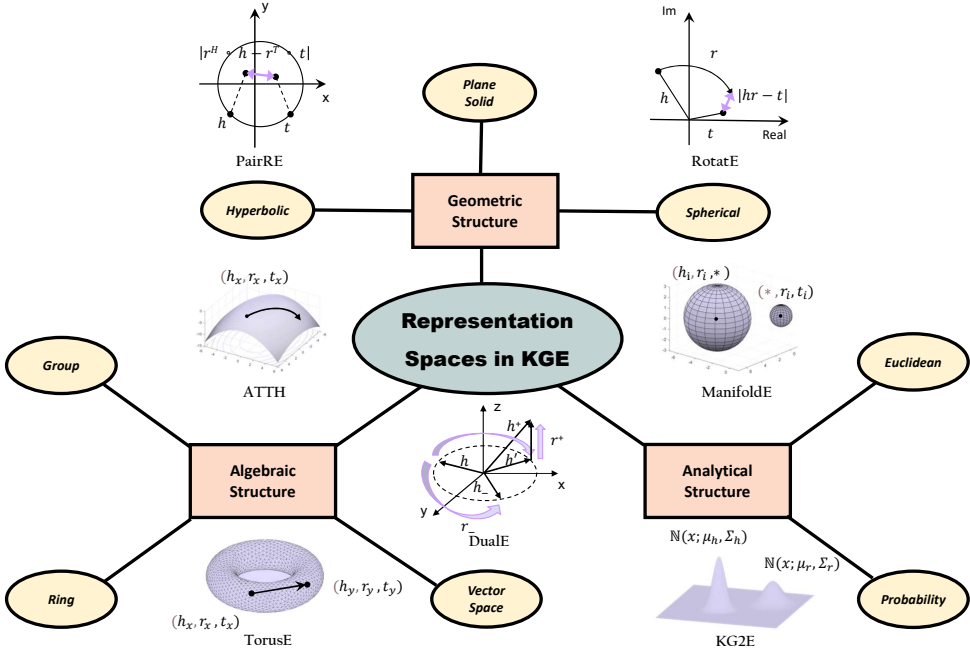


Fig. 2. Three perspectives and corresponding instances for introducing representation spaces in knowledge graph embedding: (a) Algebraic Structure. (b) Geometric Structure. (c) Analytical Structure.

generating statements that are not factually correct. (2) Since LLMs are pretrained on some general domain corpus, they may not generalise well on some domain-specific tasks, such as biomedical tasks. (3) LLMs are black-box models and it is difficult for them to provide sufficient explainability for their predictions, which is critical in some medical tasks [131]. In comparison, KGs store factual knowledge about the real world in a structured way and may provide a potential solution to address the shortcomings of LLMs. Indeed, there are some recent works that leverage KGs to enhance the performance and explainability of LLMs. In order to alleviate the hallucination issue, some work [143, 152, 187, 232] propose to incorporate KGs into the pretraining of LLMs to encode factual knowledge. For example, the KEPLER [187] proposes to learn LLMs and KGE in a unified manner, which is achieved by using a combination of KGE objective and masked language modelling objective for training. Moreover, there are some works [107, 120, 215, 228] that leverage domain-specific KGs to enhance the performance of LLMs on some domain-specific tasks. For example, the MoP model [120] infuses the biomedical knowledge stored in the biomedical knowledge graph UMLS [10] into different BERT models to enhance their performance on several downstream biomedical tasks. Furthermore, there are also some works that leverage KGs to improve the explainability of LLMs. One line of research focuses on using KGs for LLM probing, which aims to understand the relational knowledge stored in the LLMs [80, 121, 137]. Particularly, LAMA [137] is the first work to probe the relational knowledge in BERT using KGs. LAMA evaluates the knowledge in BERT by converting facts in KGs into cloze statements and using LLMs to predict the missing entity. Therefore, we believe KGs and KGEs are still useful in the context of LLMs since they can be complementary to LLMs, which is helpful in improving the performance and explainability of LLMs [131].

To the best of our knowledge, we are the first survey to summarise KGE models by establishing a comprehensive mathematical spatial architecture. To sum up, the contributions of our work can be summarised as follows:

- This is the first paper that comprehensively surveys the relationships between mathematical spaces and KGE techniques. Particularly, we summarise properties of different mathematical spaces used in KGE methods, so as to clearly understand their mathematical properties for different KGE approaches.
- We categorise existing KGE models according to their representation spaces, while providing detailed descriptions and comparisons of these works from the perspective of mathematical spaces.
- We provide ideas of space selection for the KGE task based on our analysis on the essential properties of different spaces, which could help researchers and practitioners better understand the space characteristics, and provide guidance for building their KGE models (including loss function, optimisations, etc.).
- We put forward some suggestions and future directions for the KGE tasks by showing some unique properties in different mathematical spaces/structures. These properties can be inspired and generalised to other scenarios such as natural language processing, computer vision, etc., not only for the KGE task.

The remainder of this article is organised as follows. Section 2 introduces notations and the rigorous definitions of fundamental mathematical spaces, as well as the relationships between them. This section will provide some preliminary knowledge about various representation spaces, and build connection between these spaces and the three key components of KGE models (i.e. embedding mapping, score function and representation training). Since the fundamental mathematical spaces could not cover diverse spaces used in existing KGE methods, we develop a systematic and comprehensive framework to categorise KGE methods from the perspective of representation space. To highlight the excellent effect that different mathematical features could give to KGE, section 3 introduces the proposed classification category, properties of different spaces, as well as summarise how spatial advantages work in KGE models. Subsequently, Section 4 introduces some spatially related KG downstream tasks. Through the results, the advantages of mathematical space in particular scenes and which features are critical to the tasks are well summarised. Finally, we present our conclusion and future work in Section 5, in which we summarise the respective strengths of three different mathematical structures and the reasons behind them, which will help inspire us to construct state-of-the-art algorithms in more fields, no limited to KGE.

2 Preliminaries

In this section, we introduce the notations used throughout the paper, provide two lines of preliminary knowledge related to our survey: knowledge graph embedding (KGE) and fundamental mathematical spaces. Specifically, we introduce our notations in subsection 2.1, provide an overview of knowledge graph embedding methods in subsection 2.2 and briefly introduce some basic mathematical spaces and their relationships in subsection 2.3.

2.1 Notations and Mathematical Background

The definitions of some mathematical terminologies and symbols that appear in the text are shown in Table 1. We denote mathematical spaces by blackboard bold characters (e.g., \mathbb{S} denotes topological space). Particularly, we use \mathbb{R} and \mathbb{C} to denote the field of real numbers and the field of complex numbers, respectively. We represent scalars with normal characters (e.g., $x \in \mathbb{R}$ denotes a real scalar), while vectors and matrices are denoted by the bold lowercase characters (e.g., $\mathbf{z} \in \mathbb{R}^n$

Table 1. Descriptions of symbols and terminologies.

Symbol or Terminology	Definition
\emptyset	The set containing no elements is called an empty set \emptyset .
$\mathbb{R}^n, \mathbb{C}^n \dots$	Commonly, \mathbb{R}^n represents n -dimensional (Real) Vector Space, \mathbb{C}^n denotes n -dimensional Complex Vector Space. Other spaces' symbols will be explained in additional detail below.
\mathbb{S}, τ	A set \mathbb{S} is a finite or infinite collection of objects in which order has no significance, and multiplicity is generally also ignored. And the set τ is an <i>open set</i> if every point in τ has a neighbourhood lying in the set.
Intersection(\cap), union(\cup)	The intersection of two sets \mathcal{A} and \mathcal{B} is the set of elements common to \mathcal{A} and \mathcal{B} ($\mathcal{A} \cap \mathcal{B}$). The union of two sets \mathcal{A} and \mathcal{B} is the set obtained by combining the members of each ($\mathcal{A} \cup \mathcal{B}$).
Field	A field is any set of elements that satisfies the field axioms for both addition and multiplication and is a commutative division algebra.
Complete(ness)	A space is a complete metric space in which every Cauchy sequence is convergent.
Complex conjugate	The complex conjugate of a complex number $z = a + bi$ is defined to be $\bar{z} = a - bi$.
Homeomorphism	A homeomorphism, also called a continuous transformation, is an equivalence relation and one-to-one correspondence between points in two geometric figures or topological spaces that is continuous in both directions.

denotes a real vector) and the bold uppercase characters (e.g., $\mathbf{X} \in \mathbb{R}^{m \times n}$ denotes a real-valued matrix), respectively.

We represent a knowledge graph as $\mathcal{G} = \{\mathcal{E}, \mathcal{R}, \mathcal{T}\}$, where \mathcal{E} denotes the set of entities (nodes), \mathcal{R} denotes the set of relations (the types of edges) and \mathcal{T} represents the relational facts (edges) in the knowledge graph. Facts observed in \mathcal{G} are stored as a collection of triples: $\mathcal{T} = \{(h, r, t)\}$, where each triple consists of a head entity $h \in \mathcal{E}$, a tail entity $t \in \mathcal{E}$, and a relation $r \in \mathcal{R}$ between them, e.g., $\langle \textit{Beijing}, \textit{isCapitalOf}, \textit{China} \rangle$ represents the fact that Beijing is the capital of China. We also use lower case and bold character to denote the embeddings of entities and relations. Specifically, for a fact triple (h, r, t) , we represent the embeddings of head entity, tail entity and the relation between them as \mathbf{h} , \mathbf{t} and \mathbf{r} , respectively.

2.2 Knowledge Graph Embedding

Given a knowledge graph \mathcal{G} , the goal of KGE is to learn a mapping function f , which projects the entities and relations in \mathcal{G} into a dense and low-dimensional space. The learned embeddings are expected to preserve the structural and attribute information of the original knowledge graph as much as possible, such that they can be leveraged to effectively and efficiently infer the relationships between entities.

Normally, the paradigm of learning knowledge graph embeddings consists of three components, namely *embedding mapping*, *score function* and *representation training*. The *embedding mapping component* projects the entities and relations to a low-dimensional space and represents them as embedding vectors. A variety of mathematical spaces have been used to define the embeddings of entities and relations. For example, TransE [14] proposes to learn embeddings in Euclidean space to model the transformation of entities, while RotatE [158] proposes to embed KGs into complex vector

space to model symmetry/antisymmetry of relations. Additionally, many other mathematical spaces have also been leveraged in KGEs, such as probability space [65, 197], hyperbolic space [4, 132] and spherical space [21, 37, 198]. The main focus of this survey is to provide an thorough review of KGE methods from the perspective of embedding space, summarising different properties of different embedding spaces and providing guidance for building KGE methods.

The *score function* is another key component of KGE methods. Score function, denoted as $s(\cdot)$, is a function defined on the mapped embedding space, which is used to measure the plausibility that a triple holds. Specifically, the score function is defined on the embedding vectors of a triple, i.e., $s(\mathbf{h}, \mathbf{r}, \mathbf{t})$, which is supposed to assign higher scores to positive triples (real facts) while assigning lower scores to negative triples (false facts). Based on different representation spaces, different score functions have been proposed to measure the plausibility of triples [14, 158, 198]. For example, in Euclidean representation space, the translation-based score function $s(\mathbf{h}, \mathbf{r}, \mathbf{t}) = -\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|_{1/2}$ [14] is widely used to measure the confidence that a triple is positive. Based on the different properties of representation spaces, different score functions can be defined. Please refer to Section 3 for detailed discussion of different score functions.

The *representation training* component of KGE methods aims to learn the entity and relation embeddings by maximising the scores of positive triples while minimising the scores of negative triples. Since it is infeasible to obtain the precise positive and negative triples in a knowledge graph, one convention in the field of KGE is to regard observed triples, i.e., \mathcal{T} , as positive examples while sampling unobserved triples, i.e., \mathcal{T}^- , as negative examples [14, 190]. The negative triples can be generated by randomly replacing the head entity or the tail entity in an observed triple with a random entity sampled from the entity set, i.e.,

$$\begin{aligned} \mathcal{T}^- = & \{(h^-, r, t) \mid (h, r, t) \in \mathcal{T} \wedge (h^-, r, t) \notin \mathcal{T} \wedge h^- \in \mathcal{E}\} \cup \\ & \{(h, r, t^-) \mid (h, r, t) \in \mathcal{T} \wedge (h, r, t^-) \notin \mathcal{T} \wedge t^- \in \mathcal{E}\}. \end{aligned} \quad (1)$$

Given the positive and negative triples (i.e. \mathcal{T} and \mathcal{T}^-), various objective functions can be used to train representations of KGE models. For example, the *margin-based ranking loss* [14, 75, 109, 190] is a widely adopted objective function for training representations of entities and relations. It is defined as:

$$\mathcal{L}_{margin} = \sum_{(h,r,t) \in \mathcal{T}} \sum_{(h^-,r,t^-) \in \mathcal{T}^-} \max(0, \gamma - s(\mathbf{h}, \mathbf{r}, \mathbf{t}) + s(\mathbf{h}^-, \mathbf{r}, \mathbf{t}^-)), \quad (2)$$

where γ is a margin hyperparameter. The margin-based ranking loss assumes that observed triples are more valid than unobserved triples, and therefore favours higher scores of observed triples \mathcal{T} than unobserved triples \mathcal{T}^- . Another widely used objective function for learning knowledge graph embeddings is the *cross-entropy loss* [33, 85, 158, 168], which is defined as:

$$\mathcal{L}_{CE} = \sum_{(h,r,t) \in \mathcal{T} \cup \mathcal{T}^-} \log(1 + \exp(-y_{hrt} \cdot s(\mathbf{h}, \mathbf{r}, \mathbf{t}))), \quad (3)$$

where $y_{hrt} \in \{-1, 1\}$ is the label of a triple (h, r, t) , and $y_{hrt} = 1$ indicates that the triple is a positive example while $y_{hrt} = -1$ indicates that the triple is a negative example. After defining the objective function, the knowledge graph embeddings are learned by minimising the objective function via stochastic optimisation, where a small batch of positive examples and negative examples are sampled for optimisation at each training iteration. It is worth noting that most KGE models [14, 129, 154, 168, 209] have additional constraints that the norm of knowledge graph embeddings is less than or equal to 1. Such constraints can prevent the model to trivially minimise the objective function by simply increasing the norm of the embedding vectors [14].

2.3 Fundamental Mathematical Spaces

In general, various types of fundamental mathematical spaces have been widely used for representation learning in computer science. However, there is no easy way to elaborate all these spaces due to the complex inclusion and overlapping of these spaces. Here we first introduce some basic spaces through algebraic definitions. We start by introducing the topological space, which is the most basic mathematical space. *Topological Space* is a kind of mathematical structure defined on a set, from which the concept of *convergence*, *connectivity*, *continuity* etc., are introduced [145]. A special case of topological space is the *Metric Space*, where the *distance* between any elements in the set is introduced in the topological space. The *Vector Space*, also referred to as *Linear Space* [145], is another fundamental space that are widely used in representation learning. The vector space is a set that defines both *addition* and *multiplication* satisfying eight operation conditions (see § 2.3.2 for details). On the basis of vector space, *Normed Vector Space* is defined, which additionally introduces the concept of *norm* to define the *length* between elements in the set. Particularly, *Inner Product Space* is a special normed vector space, which additionally introduce *inner product* to define the concept of *angle* between elements in the set. *Euclidean Space* is a finite dimensional inner product space which is widely used nowadays [87, 145].

The inclusion relationship of the above spaces can be summarised by: $\{Inner\ Product\ Vector\ Space\} \subset \{Normed\ Vector\ Space\} \subset \{Metric\ Space\} \subset \{Topological\ Space\}$. Going from the left to the right, each category is carried by the next one. In inner product vector space, we can use *inner product* to express both the *length* and *angle* of vectors, since the *inner product* induces a *norm*. But in a normed vector space, we could just measure the distance between two points through space metric. Then in metric space and topological space, there is less basic concept that can be measured. In another word, the “capacity” of spaces is getting weaker. To sum up, the *inner product* induces *norm*, *norm* induces distance, and distance induces topology, therefore: *every inner product space is a normed vector space, every normed vector space is a metric space and every metric space is a topological space*. It should be noticed that linear space is an algebraic structure while topological space is a topological structure. So they are not juxtaposed since they are considered in different categories. The relationships between different mathematical spaces are presented in Figure 3. In what follows, we provide rigorous definitions of above mathematical spaces, which also be summarised in Table 2.

2.3.1 Topological Space

Definition 2.1. A *topological space* is a set \mathbb{S} in which a collection of subsets τ (called *open sets*, see Table 1) is specified, with the following properties [145]: \mathbb{S} is open, \emptyset is open, the intersection (see Table 1) of any two open sets is open, and the union of every collection of open sets is open. Such a collection τ is called a *topology* on \mathbb{S} . When clarity seems to demand it, the topological space corresponding to the topology τ will be written as (\mathbb{S}, τ) rather than \mathbb{S} . Metric space, uniform space [90, 167] are examples of topological space.

2.3.2 Vector Space (Linear Space)

Definition 2.2. Let Φ stands for either \mathbb{R} or \mathbb{C} , i.e., real field or complex field (See Table 1). A *scalar* is a member of the *scalar field* Φ . A *vector space* over Φ is a set \mathbb{X} , whose elements are called vectors, and in which two operations, *addition* and *scalar multiplication*, are defined, with the following algebraic properties [145]:

- (a) For any two vectors \mathbf{x} and \mathbf{y} in \mathbb{X} , the addition between them is represented as $\mathbf{x} + \mathbf{y}$, with the following properties:

$$\mathbf{x} + \mathbf{y} = \mathbf{y} + \mathbf{x} \text{ and } \mathbf{x} + (\mathbf{y} + \mathbf{z}) = (\mathbf{x} + \mathbf{y}) + \mathbf{z}. \quad (4)$$

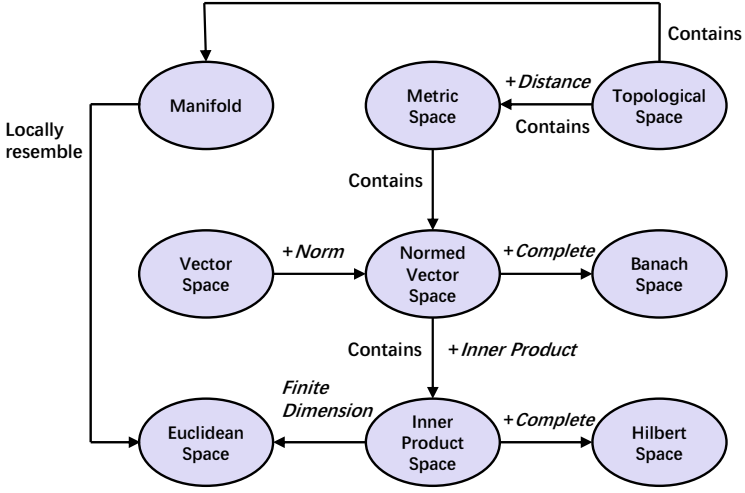


Fig. 3. The relationships between different mathematical spaces. The basic spatial relationship is $\{\text{Inner Product Vector Space}\} \subset \{\text{Normed Vector Space}\} \subset \{\text{Metric Space}\} \subset \{\text{Topological Space}\}$. In addition, we tease out more detailed relations, such as the *complete* case of Inner Product Space is Hilbert Space.

The vector space \mathbb{X} contains a unique vector $\mathbf{0}$ (the *zero vector* or *origin* of \mathbb{X}) such that $\mathbf{x} + \mathbf{0} = \mathbf{x}$ for every $\mathbf{x} \in \mathbb{X}$, and each $\mathbf{x} \in \mathbb{X}$ corresponds a unique vector $-\mathbf{x}$ such that $\mathbf{x} + (-\mathbf{x}) = \mathbf{0}$.

- (b) For any scalar $\alpha, \beta \in \Phi$ and vector $\mathbf{x} \in \mathbb{X}$, the scalar multiplication between them is denoted as $\alpha\mathbf{x}$, with the following properties:

$$\alpha(\mathbf{x} + \mathbf{y}) = \alpha\mathbf{x} + \alpha\mathbf{y}, (\alpha + \beta)\mathbf{x} = \alpha\mathbf{x} + \beta\mathbf{x}. \quad (5)$$

Like the zero vector in vector space, the zero element of the scalar field can also be defined in a similar way, which is denoted as 0.

Particularly, a *real vector space* is the one for which $\Phi = \mathbb{R}$, while a *complex vector space* is the one for which $\Phi = \mathbb{C}$. In the following, any statements about vector spaces in which the scalar field is not explicitly specified means that they can be applied to both real vector space and complex space.

2.3.3 Normed Space

Definition 2.3. A vector space \mathbb{X} is said to be a *normed space* if each vector $\mathbf{x} \in \mathbb{X}$ is associated with a nonnegative real number $\|\mathbf{x}\|$, called the *norm* of \mathbf{x} [145], in such a way that:

- $\|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$ for all \mathbf{x} and \mathbf{y} in \mathbb{X} ,
- $\|\alpha\mathbf{x}\| = |\alpha| \|\mathbf{x}\|$ if $\mathbf{x} \in \mathbb{X}$ and α is a scalar,
- $\|\mathbf{x}\| > 0$ if $\mathbf{x} \neq \mathbf{0}$.

The word “norm” is also used to denote the *function* that maps \mathbf{x} to $\|\mathbf{x}\|$. Every normed space can be regarded as a particular metric space, in which the distance $d(\mathbf{x}, \mathbf{y})$ between \mathbf{x} and \mathbf{y} is defined as $\|\mathbf{x} - \mathbf{y}\|$. The properties of the distance function d in metric space are:

- $0 \leq d(\mathbf{x}, \mathbf{y}) < \infty$ for all \mathbf{x} and \mathbf{y} ,
- $d(\mathbf{x}, \mathbf{y}) = 0$ if and only if $\mathbf{x} = \mathbf{y}$,
- $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$ for all \mathbf{x} and \mathbf{y} ,
- $d(\mathbf{x}, \mathbf{z}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z})$ for all $\mathbf{x}, \mathbf{y}, \mathbf{z}$.

Table 2. Descriptions of four basic spaces in mathematics: Vector Space, Normed Space, Inner Product Space and Topological Space. It is important to note that they are listed here for a general introduction, not because they are juxtaposed. The specific relationships between these spaces can found in §2.3.

Space	Description	Property
<i>Topological Space</i>	a geometrical space in which closeness is defined.	<ul style="list-style-type: none"> • <i>open set</i>
<i>Vector Space</i>	a set of vectors in which two operations, addition and scalar multiplication are defined.	<ul style="list-style-type: none"> • <i>addition</i> • <i>multiplication</i>
<i>Normed Space</i>	a vector space over the real or complex numbers, on which a norm is defined.	<ul style="list-style-type: none"> • <i>addition</i> • <i>multiplication</i> • <i>norm: $\ x\$</i>
<i>Inner Product Space</i>	a vector space with a binary operation called an inner product .	<ul style="list-style-type: none"> • <i>addition</i> • <i>multiplication</i> • <i>inner product: $\langle x, x \rangle$</i>
<i>Euclidean Space</i>	a commonly used finite dimensional inner product space over real numbers.	<ul style="list-style-type: none"> • <i>addition</i> • <i>multiplication</i> • <i>inner product: $\langle x, x \rangle$</i>
<i>Manifold</i>	a topological space which is locally Euclidean .	<ul style="list-style-type: none"> • each point has a certain neighbourhood

A *Banach Space* is a special normed space which is *complete* (see Table 1) in the metric defined by its norm, which means that every Cauchy sequence is required to converge. (A Cauchy sequence is a sequence whose elements become arbitrarily close to each other as the sequence progresses [87].)

2.3.4 Inner Product Space

Definition 2.4. A complex vector space \mathbb{H} is called an *inner product space* or *unitary space* if each ordered pair of vectors \mathbf{x} and \mathbf{y} in \mathbb{H} is associated with a complex number $\langle \mathbf{x}, \mathbf{y} \rangle$, called the *inner product* or *scalar product* of \mathbf{x} and \mathbf{y} , such that the following rules hold [145]:

- $\langle \mathbf{y}, \mathbf{x} \rangle = \overline{\langle \mathbf{x}, \mathbf{y} \rangle}$ (The bar denotes complex conjugation (see Table 1)),
- $\langle \mathbf{x} + \mathbf{y}, \mathbf{z} \rangle = \langle \mathbf{x}, \mathbf{z} \rangle + \langle \mathbf{y}, \mathbf{z} \rangle$,
- $\langle \alpha \mathbf{x}, \mathbf{y} \rangle = \alpha \langle \mathbf{x}, \mathbf{y} \rangle$ if $\mathbf{x} \in \mathbb{H}$, $\mathbf{y} \in \mathbb{H}$, $\alpha \in \mathbb{C}$,
- $\langle \mathbf{x}, \mathbf{x} \rangle \geq 0$ for all $\mathbf{x} \in \mathbb{H}$,
- $\langle \mathbf{x}, \mathbf{x} \rangle = 0$ only if $\mathbf{x} = 0$.

Particularly, if the normed space is complete, it is called a *Hilbert Space*.

2.3.5 Euclidean Space

Definition 2.5. *Euclidean space* is the basic space of geometry, intended to represent physical space [6]. Generally, Euclidean space refers to Euclidean vector space, which is a finite dimensional inner product space over real numbers. Based on the algebraic definition of Euclidean space, a plane or solid (i.e. Euclidean geometry) can be well represented by lines and points.

2.3.6 Manifold

Definition 2.6. A topological space \mathbb{M} is said to be a *manifold* or *locally Euclidean of dimension n* if every point of \mathbb{M} has a neighbourhood in \mathbb{M} that is homeomorphic (see Table 1) to an open subset of

n -dimensional Euclidean space [90]. Manifolds constitute a generalisation of objects and the concept of a manifold is central to many parts of geometry since it allows complicated structures, such as sphere, curved surface, etc. to be described in terms of well-understood topological properties of simpler spaces.

3 Representation Spaces in Knowledge Graph Embedding

Since a KG usually consists of many complicated structures (e.g., 1-to-N, N-to-N, and hierarchical relationships), researchers have proposed to embed KGs in different representation spaces in order to better preserve such complicated structural information [24, 38, 59, 198, 225]. Indeed, different representation spaces have their unique structures and properties, as we show in Section 2. However, in addition to the fundamental mathematical spaces introduced in Section 2, there are more spaces that provide better properties for KGE. For example, in hyperbolic space, the region and length increase exponentially with the radius, which provides more available space for embedding task [4, 24, 101, 102, 126, 195]. Moreover, in Lie group, embedding vectors will never diverge unlimitedly and therefore regularisation of embedding vectors is no longer required for effective learning [38]. As a result, KGE methods built on different representation spaces are able to capture and preserve different structural and attribution information in original KGs. However, there is neither a systematic review of KG embedding methods from the perspective of representation spaces, nor any literature showing how to properly choose representation space given particular KGE tasks. In this paper, we aim to fill this gap by summarising KGE methods based on the structures and properties of their mathematical representation spaces.

Note that in Section 2, we introduced some algebraic definitions of some basic spaces. Based on them, some geometric perspectives, such as Euclidean geometry and hyperbolic geometry, can also be introduced accordingly. At the same time, we notice that there are various kinds of mathematical spaces in KGE, which play a significant role and belong to different mathematical structures. The diverse spaces in KGE often have complicated relationships. For example, manifolds and Euclidean geometry have inclusion relations since there are overlapping structures between them. In addition, some spaces are not juxtaposed such as spherical space and probability space because they originate from different mathematical structures, which cannot be discussed in the same category.

As a result, in order to better understand the influence of different mathematical representation spaces on KGE methods, we build a systematic, comprehensive, multi-angle categorisation to classify the special spaces and categorise KGE models more accurately based on three mathematical structures, namely **Algebraic Structure**, **Geometric Structure** and **Analytical Structure**. Most KGE models fall under these three structures, which proves the rationality of our categorisation, as shown in Figure 4. It is worth noting that we will add additional definitions in Section 3 for KGE spaces that are not mentioned in Section 2.

In this section, we will first describe the definitions and properties of the above three mathematical structures, after which we will provide some representative KGE methods that learn embedding in the corresponding mathematical structure, as well as summarise how spatial advantages work in KGE models.

3.1 Algebraic Structure

An algebraic structure is a nonempty set on which one or more finite operations satisfying the axiom are defined [56, 192]. Some representative algebraic structures include *vector space*, *group* and *ring*. For example, the vector space \mathbb{X} is an algebraic structure that involves many common *binary operations*, such as addition, subtraction and multiplication. From the algebraic point of view, most models in machine learning and even knowledge graph embedding involve algebraic

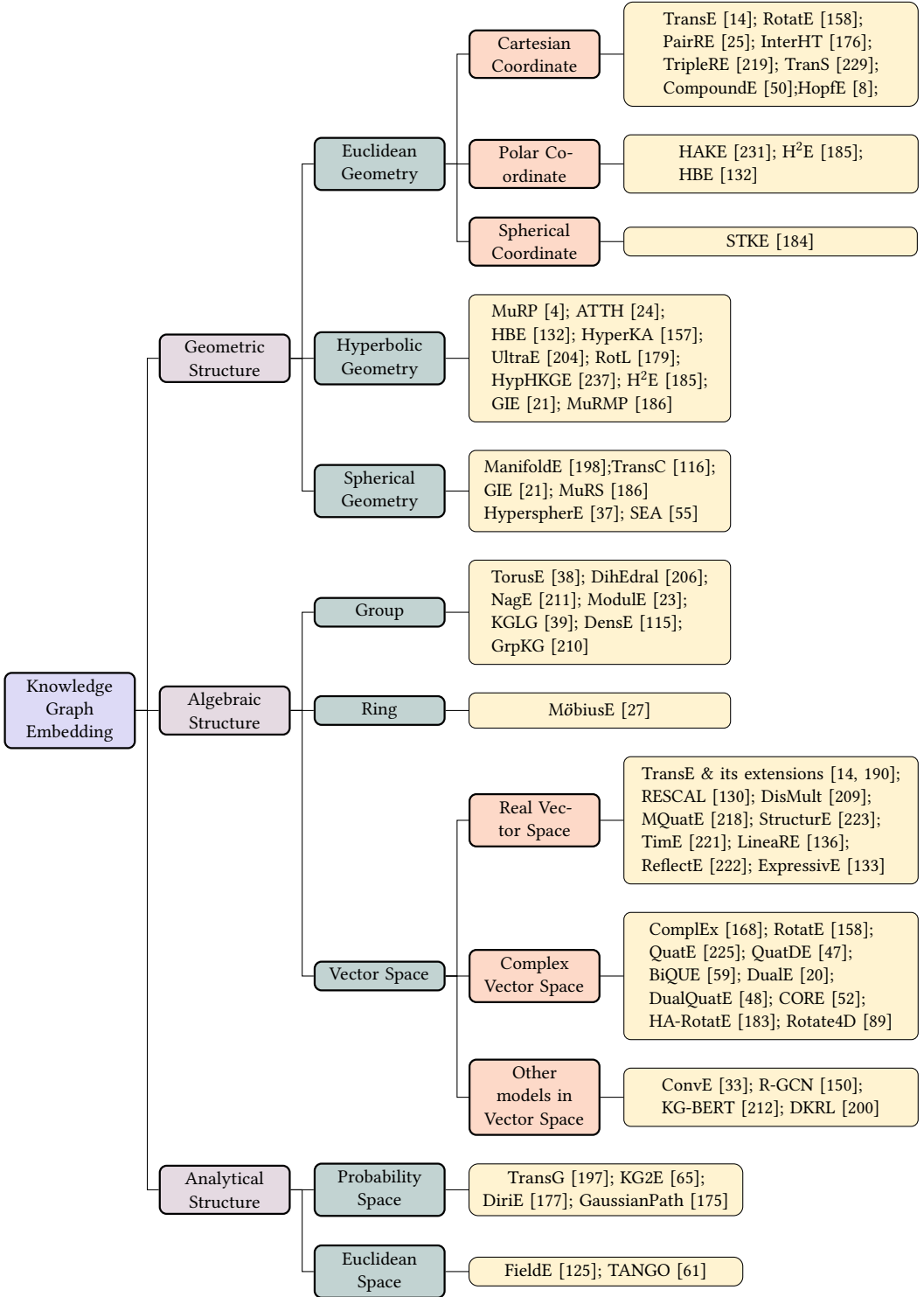


Fig. 4. The systematic categorisation of KGE models based on three mathematical perspectives.

operations, such as möbius embedding [4, 27, 169], group embedding [38, 115], user and word embedding [101, 102, 195] etc., which all belong to the methods in the category of algebraic structure.

3.1.1 Vector Space

The **Vector Space** is the most widely adopted mathematical space in the field of machine learning, whose definition is provided in Section 2.3.2. Since there are two convenient algebraic operations defined in vector space: *vector addition* and *scalar multiplication*, it is also widely used by many KGE methods. These methods utilise these operations to project both entities and relations into the same vector space, with the objective of preserving the relational interactions among entities in the vector representation space. As discussed in Section 2.3.2, the vector space can be classified as *real vector space* and *complex vector space* based on the scalar field. In accordance with these classification criteria, we categorise vector space-based KGE methods into three distinct groups: real vector space-based, complex vector space-based, and other models within the vector space domain, such as neural network-based and external information-based models. In what follows, we delve into each of these groups of KGE methods in depth.

Real Vector Space. One representative KGE methods based on real vector space is TransE [14]. Given a knowledge graph, it projects all entities and relations into a low-dimensional real vector space \mathbb{R}^k . Specifically, it directly represents the head entity h , the relation r and the tail entity t in a fact triple as embedding vectors $\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^k$, respectively. If the triple (h, r, t) holds, the embedding of the tail entity \mathbf{t} should be as much as close to the head entity embedding \mathbf{h} translated by the relation embedding \mathbf{r} , which conforms to the principal: $\mathbf{h} + \mathbf{r} \approx \mathbf{t}$. It is very simple and effective to directly express the relations between entities by *addition* operation. Therefore, the score function in TransE is defined as:

$$s(\mathbf{h}, \mathbf{r}, \mathbf{t}) = -\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|_{1/2}. \quad (6)$$

However, despite TransE is simple and effective, it performs poorly when dealing with multi-relation (e.g., 1-to- N , N -to-1, N -to- N) data [165]. For example, suppose that a relation r_1 is a 1-to- N relation, then for a head entity h_1 , it may have relation r_1 with two different tail entities t_1 and t_2 . TransE would enforce these two tail entities to have approximately the same embedding, i.e., $\mathbf{t}_1 \approx \mathbf{t}_2$, which is inaccurate since t_1 and t_2 are two different entities. The same analysis also applies to N -to-1 and N -to- N relations. To overcome the problems of TransE in modelling multi-relation data, TransH [190] projects the head entity h and tail entity t into the hyperplane where the relationship r resides. Specifically, TransH assumes that each relation embedding \mathbf{r} lies in a different relation-specific hyperplane \mathbf{w}_r . In order to measure the plausibility that a triple (h, r, t) holds, the head entity embedding \mathbf{h} and tail entity embedding \mathbf{t} are first projected into the relation-specific hyperplane \mathbf{w}_r :

$$\mathbf{h}_\perp = \mathbf{h} - \mathbf{w}_r^\top \mathbf{h} \mathbf{w}_r, \mathbf{t}_\perp = \mathbf{t} - \mathbf{w}_r^\top \mathbf{t} \mathbf{w}_r, \quad (7)$$

where \mathbf{h}_\perp and \mathbf{t}_\perp represent the projected head and tail entity embeddings, respectively. Therefore, the score function of a triple (h, r, t) is defined as:

$$s(\mathbf{h}, \mathbf{r}, \mathbf{t}) = -\|\mathbf{h}_\perp + \mathbf{r} - \mathbf{t}_\perp\|_2^2. \quad (8)$$

By projecting the entity embedding to relation-specific hyperplane, TransH allows entities to have different embeddings in different relations, which ensures that the embedding of t is different even if the head entity or the relation is the same. From TransH, we can see that *projection* plays a key role, and projection is a very common operation in vector space that is used to establish various connections (See Figure 5a).

Despite TransH can effectively handle multi-relations in KGs, it still assumes that the relation embeddings and entity embeddings belong to the same embedding space, which will limit diversity. However, an entity may have multiple aspects and different relations may focus on different aspects of entities. To address this problem, TransR [109] proposes to embed entities in an entity space \mathbb{R}^k ($\mathbf{h}, \mathbf{t} \in \mathbb{R}^k$), and embed relations in a different relation space \mathbb{R}^d ($\mathbf{r} \in \mathbb{R}^d$, k and d are not necessarily identical). The entity embedding and relation embedding are correlated by the relation-specific mapping matrix ($\mathbf{M}_r \in \mathbb{M}^{k \times d}$), which projects entity embedding from entity space to relation space. Therefore, the score function in TransR is defined as:

$$s(\mathbf{h}, \mathbf{r}, \mathbf{t}) = -\|\mathbf{M}_r \mathbf{h} + \mathbf{r} - \mathbf{M}_r \mathbf{t}\|_2^2. \quad (9)$$

RESCAL [130] models the semantic interactions between entities by using bilinear operations with score function:

$$s(\mathbf{h}, \mathbf{r}, \mathbf{t}) = \mathbf{h}^\top \mathbf{M}_r \mathbf{t} = \left(\sum_i \sum_j [\mathbf{M}_r]_{ij} \cdot \mathbf{h}_i \cdot \mathbf{t}_j \right), \quad (10)$$

where each relation is defined as a matrix $\mathbf{M}_r \in \mathbb{M}^{d \times k}$, and $[\mathbf{M}_r]_{ij}$ denotes the i -th row and j -th column of the matrix \mathbf{M}_r . Similar linear models such as DisMult [209], HoLE [129], ANALOGY [110], Simple [82], TuckER [5] and LowFER [2] have proven their great performance on downstream tasks. As a result, with simple and efficient linear operations, the plausibility of facts can be measured by matching latent semantics of entities and relations (See Figure 5b).

Recently, there are some KGE models that learn embeddings in the real vector space as well. For example, ReflectE [222] uses reflection transformation to map properties and entities by Householder matrix. LineaRE [136] interprets a relation as a linear function of entities to capture connectivity patterns. TimE [221] projects entities into the nonlinear time domain to obtain better diversity distribution. It could be found that many mathematical operations of real vector space are worth exploring and applying to KGE.

Complex Vector Space. Compared with real vector embedding, complex vector embedding (or complex embedding for brevity) can handle a large variety of binary relations [168], such as symmetric and antisymmetric. Complex embedding methods for KGs, which embed entities and relations in the **Complex Vector Space** (i.e., $\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{C}^k$), have also been widely studied. ComplEx [168] is the first model to use complex embedding in KGE. Specifically, it defines a score function with the Hermitian dot product in complex vector space:

$$s(\mathbf{h}, \mathbf{r}, \mathbf{t}) = \text{Re}(\mathbf{h}^\top \text{diag}(\mathbf{r}) \bar{\mathbf{t}}) = \text{Re} \left(\sum_i \mathbf{r}_i \cdot \mathbf{h}_i \cdot \mathbf{t}_j \right), \quad (11)$$

where $\bar{\mathbf{t}}$ is the conjugate (see Table 1) of \mathbf{t} , $\text{Re}(\cdot)$ denotes the operation to obtain the real part of a complex number. Since the score function is not symmetric anymore, the facts with antisymmetric relations can receive different scores depending on the ordering of entities. Thus ComplEx can effectively capture antisymmetric relations while retaining the efficiency benefits of the dot product.

Motivated by the Euler's rule $e^{i\theta} = \cos\theta + i\sin\theta$, RotatE [158] maps entities and relations into the complex vector space and defines each relation as a rotation from the source entity to the target entity with the principal $\mathbf{t} = \mathbf{h} \circ \mathbf{r}$ (where \circ denotes the Hadamard product, i.e., element-wise product). Different entities can be directly modelled through the angular transformation (See Figure 5c) so as to capture some patterns including symmetry, antisymmetry, inversion, and composition. Compared with ComplEx, QuatE [225] takes advantage of quaternion representations to enable richer and more expressive semantic matching between head and tail entities with the use of Hamilton product (\otimes , i.e., quaternion multiplication), where a quaternion Q is defined as

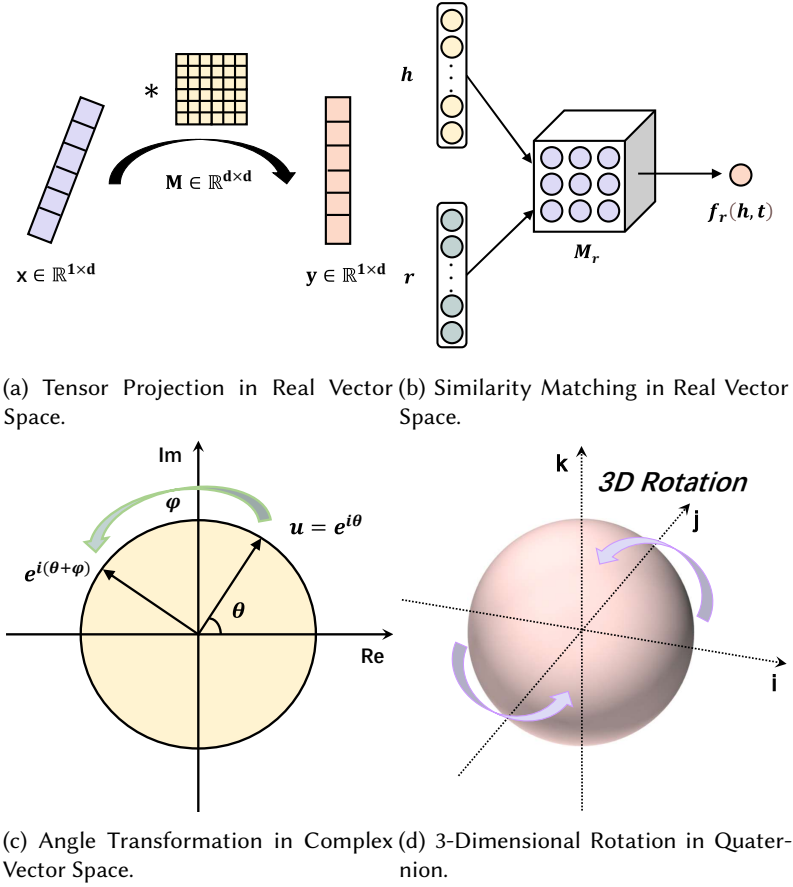


Fig. 5. An Illustration of some kind of algebraic operations which can be applied for knowledge graph embedding from algebraic perspective.

$Q = a + bi + cj + dk$ and (i, j, k) are imaginary units satisfying Hamilton's rule: $i^2 = j^2 = k^2 = ijk = -1$. BiQUE [59] extends the quaternion system to a more powerful algebraic system called biquaternion $q = (w_r + w_i\mathbf{I}) + (x_r + x_i\mathbf{I}) + (y_r + y_i\mathbf{I}) + (z_r + z_i\mathbf{I})$, where $w_r, x_r, y_r, z_r, w_i, x_i, y_i, z_i \in \mathbb{R}$. Also by utilising the Hamilton product of biquaternions, BiQUE imbues itself with a strong geometric interpretation (i.e., the Euclidean/hyperbolic rotation). In addition to the fact that quaternions have more degree of freedom in the four dimensional space. It is worth mentioning that the interpolation between two quaternions is extremely easy, which helps to establish rotations (See Figure 5d). Recently, DualE [20] employs a novel framework which can embrace both translation and rotation operations in the dual quaternion space ($Q_{dual} = a + \epsilon b$, where a and b are quaternions while ϵ is a dual unit). DualQuatE [48] combines the idea of DualE and QuatE. HA-RotatE [183] and CORE [52] inherit the structure of RotatE and expand the ability of KGs embedding.

Neural Network Models in Vector Space. Neural networks [33] are employed in KGE models for learning embedded features. However, the black-box nature of neural networks precludes our understanding of the specific mathematical features captured by these neural network-based KGE

models. Nevertheless, given that these models yield vector representations, we categorise them in conventional vector spaces.

ConvE [33] initially reshapes the head entity and relation into a 2D matrix, subsequently utilising 2D convolution in conjunction with multiple layers to represent interactions between entities and relations. The scoring function is defined as:

$$s(\mathbf{h}, \mathbf{r}, \mathbf{t}) = \sigma(\text{vec}(\sigma([\mathbf{h}_{2D}, \mathbf{r}_{2D}] * \omega))\mathbf{W})\mathbf{t}, \quad (12)$$

where σ denotes activation function, and $\text{vec}(\cdot)$ means the vectorisation operation. \mathbf{h}_{2D} and \mathbf{r}_{2D} are 2D reshaping of \mathbf{h} and \mathbf{r} (i.e., $\mathbf{h}_{2D}, \mathbf{r}_{2D} \in \mathbb{R}^{k_w \times k_h}$, if $\mathbf{h}, \mathbf{r} \in \mathbb{R}^k$ with $k = k_w k_h$), respectively. ω is the convolutional filter. Owing to the potent feature extraction capacity of the nonlinear neural network layer, ConvE exhibits high expressiveness and delivers exceptional performance.

R-GCN [150] represents relationships between entities and relations by employing Graph Convolutional Networks (GCNs [83, 213]), which focus on local graph neighbourhoods to manage large-scale relational data. To compute the forward-pass update for an entity, the hidden state \mathbf{y} of layer $l + 1$ undergoes a propagation process [213]:

$$\mathbf{y}_i^{l+1} = \sigma\left(\sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{N}_i^r} \frac{1}{c_{i,r}} \mathbf{W}_r^l \mathbf{y}_j^l + \mathbf{W}_0^l \mathbf{y}_i^l\right), \quad (13)$$

where \mathcal{N}_i^r denotes the set of neighbor indices of node i under relation $\mathbf{r} \in \mathcal{R}$ and $c_{i,r}$ is a normalisation constant. CompGCN [170] leverages a variety of composition operators derived from KGE methods to concurrently embed nodes and relations in a graph. This approach has demonstrated the capability to generalise various existing multi-relational GCN techniques, including R-GCN [150], Directed-GCN [118] and Weighted-GCN [151].

KG-BERT [212], a transformer-based [171] model, interprets triples as text sequences and conducts knowledge embedding by leveraging a pre-trained language model BERT [34]. KG-BERT is capable of utilising abundant linguistic information present in the extensive text and emphasising the most pertinent words associated with a triple. Recently, Knowformer [92] employs position-aware relational compositions to encode the semantics of entities appearing in varying positions within a relational triple and achieves state-of-the-art. It has been proved that these compositions assist the self-attention mechanism [171] in differentiating entity roles based on their positions.

Incorporate Auxiliary Information in KGE. Learning knowledge graph embeddings using auxiliary information constitutes a significant subfield within KGE approaches. While this topic may not be as closely connected to the mathematical representation space, we will provide a concise overview of some classic external-information-based KGE models to maintain a comprehensive review. Commonly used auxiliary information in existing work includes text descriptions, entity types, relational structures or paths, and other information [43, 146, 161, 182]. DKRL [200] investigates more profound knowledge representation by utilising CNNs to extract the semantics of entity descriptions in a representation learning manner. TKRL [203] considers hierarchical entity types as projection matrices and employs the type information as relation-specific type constraints. It has been demonstrated that TKRL effectively captures hierarchical type information, which is crucial for constructing representations of knowledge graphs. HRS [235] treats relations in knowledge graphs as a three-layer hierarchical relation structure, which can be effortlessly integrated into other KGE models to acquire abundant structural information. RSN [60] and Interstellar [230] tackle the issue of previous models' insufficient ability to capture long-term relational dependencies of entities effectively by employing relational paths. Recently, TransO [98] has been proposed as a method to seamlessly integrate all available ontology information (i.e., type information, relation constraint information, and hierarchical structure information.) within the knowledge embedding process,

thereby enhancing decision-making capabilities in complex situations. In addition, combined with other external information such as image data [201], conceptual information [58, 62, 72] also holds significant importance in the realm of knowledge graph embedding.

3.1.2 Group

A **group** [16, 87, 144] is an algebraic structure composed of a set and an operation, which is ubiquitous in all fields inside and outside mathematics. For example, symmetry groups describe the symmetries of geometry [142], and lie groups are used in particle physics [53]. Because of its unique abstract algebra properties, groups are also widely used in machine learning. In this section, we will start with the definition of groups and then describe the KGE models that leverage groups as embedding space.

Definition 3.1. A (binary) **operation** on a set \mathbb{G} is a function:

$$* : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}.$$

Definition 3.2. A **group** is a set \mathbb{G} equipped with an operation $*$ and a special element $e \in \mathbb{G}$, called the **identity**, such that:

- (a) (Associativity): For every $a, b, c \in \mathbb{G}$, $(a * b) * c = a * (b * c)$;
- (b) (Existence of identity): $e * a = a$ for all $a \in \mathbb{G}$;
- (c) (Existence of inverses): for every $a \in \mathbb{G}$, there is $a' \in \mathbb{G}$ with $a' * a = e$.

To tackle TransE's regularisation problem, TorusE [38] proposes to embed KGEs into a special algebraic structure—Torus. A torus is an Abelian Lie group, which is derived from the vector space through the nature projection $\pi : \mathbb{R}^n \rightarrow T^n, x \mapsto [x]$ (T^n denotes quotient space [38]). With the help of torus, the model never diverges unlimitedly.

DihEdral [206] is the first attempt to employ finite non-Abelian group in KG embedding to account for relation compositions. Since the elements in a dihedral group are well constructed by rotation and reflection operations, and the multiplication between elements can be Abelian or non-Abelian, DihEdral is capable to capture all desired properties: (skew-)symmetry, inversion and (non-)Abelian composition. DensE [115] decomposes a relation operator into a SO(3) group-based (SO(3): Special Orthogonal Group in 3 dimensions) rotation as well as a scaling transformation. NagE [211] proves for the first time that the group algebraic structure is significant for designing relational embedding models. *Specifically, the definition of group can naturally satisfy the basic properties (e.g., inversion, composition) of knowledge graphs*, which means the group-based models should have great potential to deal with KGE tasks. Other recent models based on group structure such as ModuLE [23] consider both entity and relation as group elements so as to achieve state-of-the-art performance.

3.1.3 Ring

In mathematics, **rings** [16, 87, 144] are algebraic structures that generalised fields. Commutative rings are one of the main branches of ring theory. Examples include the set of integers with addition and multiplication, and the set of polynomials with the same operations. Ring theory was later proved useful in geometry and analysis [41]. In this section, we will first introduce the definition of ring, and then discuss the KGE models that use ring as embedding space.

Definition 3.3. A **ring** is a set \mathbb{S} equipped with two binary operations: $+$ (*addition*) and \cdot (*multiplication*) which satisfy the following axioms,

- (a) \mathbb{S} is an Abelian group under addition, meaning that:
 - (Associative): $(a + b) + c = a + (b + c)$ for all $a, b, c \in \mathbb{R}$;
 - (Commutative): $a + b = b + a$ for all $a, b \in \mathbb{S}$;

- (Additive identity): There is an element 0 in \mathbb{S} such that $a + 0 = a$ for all $a \in \mathbb{S}$;
 - (Additive inverse): For each a in \mathbb{S} there exists $-a$ such that $a + (-a) = 0$.
- (b) \mathbb{S} is a monoid under multiplication, meaning that:
- (Associative): $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ for all $a, b, c \in \mathbb{S}$;
 - (Multiplicative identity): There is an element 1 in \mathbb{S} such that $a \cdot 1 = a$ and $1 \cdot a = a$ for all $a \in \mathbb{S}$.
- (c) Multiplication is distributive with respect to addition, meaning that:
- (Left distributivity): $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$ for all $a, b, c \in \mathbb{S}$;
 - (Right distributivity): $(b + c) \cdot a = (b \cdot a) + (c \cdot a)$ for all $a, b, c \in \mathbb{S}$.

MöbiusE [27] extends KGE to manifold-based embedding, in which the entities and relations are embedded to the surface of Möbius ring. With the scoring function ($dist(\mathbf{h} \oplus \mathbf{r}, \mathbf{t})$, where \oplus and $dist$ represent addition and distance function specially defined on Möbius ring, respectively), MöbiusE has much more expressiveness and flexibility than TorusE [38] due to the extra properties on ring. As MöbiusE subsumes TorusE, it naturally inherits all the desired properties of TorusE including symmetric/antisymmetry, inversion, and composition. *It is worth mentioning that the Möbius band is a non-oriented surface and the concept of clockwise and counterclockwise cannot be clearly defined, which may be helpful for certain tasks that are closely related to orientation.*

3.2 Geometric Structure

A geometric structure [54] on a manifold is a complete Riemannian metric which is locally homogeneous (i.e., any two points have isometric neighbourhoods). Although there are few detailed definitions of geometric structures, here we focus more on the knowledge graph embedding models built on different geometric models/spaces and analyse them in detail from three geometric perspectives: Euclidean Geometry, Hyperbolic Geometry and Spherical Geometry. We summarise the operations of these three geometries in Table 3.

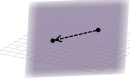
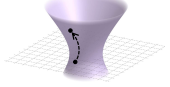
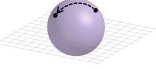
3.2.1 Euclidean Geometry

Euclidean geometry is the study of geometrical shapes (plane and solid) and figures based on different axioms and theorems. It is basically introduced for flat surfaces or plane surfaces. This part of geometry was employed by the Greek mathematician *Euclid*, who has also described it in his book, *Elements* [44]. Geometry is derived from the Greek words ‘geo’ which means earth and ‘metrein’ which means ‘to measure’.

Euclidean geometry deals with things like points, lines, angles, squares, triangles, and other shapes. Based on different coordinate systems, in this part, we will divide KGE models built in Euclidean geometry into four part: *Cartesian Coordinate*, *Polar Coordinate* and *Spherical Coordinate*. It is worth noting that: Euclidean geometry under geometric structure and vector space under algebraic structure are very similar and have some common concepts. This is because geometric features and algebraic features are usually closely related. But in this section we will emphasise the advantages of KGE models from the geometric perspective.

Cartesian Coordinate. Most of the translation-based models are based on the common Cartesian coordinates. For example, TransE follows the principle: $\mathbf{h} + \mathbf{r} \approx \mathbf{t}$, i.e., the vector of head entity add the relation vector is equal to the tail vector through a translation. These three vectors are connected from head to tail, forming a closed path in Cartesian coordinates. RotatE shows that it’s ingenious to define each relation as a rotation from the source entity to the target entity. For example, two relations \mathbf{r}_1 and \mathbf{r}_2 are inverse if and only if their embeddings are conjugate: $\mathbf{r}_1 = \bar{\mathbf{r}}_2$ ($\mathbf{r}_1 : (\cos(\theta_1), \sin(\theta_1))$, $\mathbf{r}_2 : (\cos(\theta_2), \sin(\theta_2))$), which means they are symmetric about the real axis (i.e., $\sin(\theta_1) = -\sin(\theta_2)$). All relation patterns can be clearly illustrated by geometric transformation

Table 3. Summary of operations in Euclidean, Hyperboloid and Spherical models [113, 125, 191]. It is worth noting that the category of geometric spaces associated with a model is closely related to the distance function utilised in its scoring function. Models based on Euclidean geometry typically employ the well-known Euclidean distances (i.e., $\|\cdot\|$), where varying transformation relations for \mathbf{h} , \mathbf{r} , and \mathbf{t} give rise to distinct models. Conversely, hyperbolic and spherical models tend to use hyperbolic distance ($d_{\mathbb{H}}$) and spherical distance ($d_{\mathbb{S}}$) with unique properties for their respective scoring functions. Notably, some models employing spherical embedding techniques may not exclusively rely on spherical distances. Instead, they may model entities as spheres or through other spherical geometric embeddings. Visual representations of these three geometric embeddings are included in the table.

	Euclidean	Hyperboloid	Spherical
Manifold \mathcal{M}	\mathbb{R}^n	$\mathbb{H}_K^n = \{x \in \mathbb{R}^{n+1} : \langle x, x \rangle = \frac{1}{K}\}$	$\mathbb{S}_K^n = \{x \in \mathbb{R}^{n+1} : \langle x, x \rangle = 1\}$
Distance $d(x, y)$	$\sqrt{\langle x - y, x - y \rangle}$	$\frac{1}{ K } \cosh^{-1}(K \langle x, y \rangle)$	$\cos^{-1}(\langle x, y \rangle)$
Exponential map $exp_x^K(v)$	$x+v$	$\cosh(\sqrt{ K } \ v\)x + v \frac{\sinh(\sqrt{ K } \ v\)}{\sqrt{ K } \ v\ }$	$\cos(\ v\)x + \sin(\ v\) \frac{v}{\ v\ }$
Curvature \mathfrak{C}	0	< 0	> 0
Sum of angles \mathfrak{A}	π	$< \pi$	$> \pi$
Scoring function $s(\mathbf{h}, \mathbf{r}, \mathbf{t})$	TransE: $- \mathbf{h} + \mathbf{r} - \mathbf{t} $ RotatE: $- \mathbf{h} \circ \mathbf{r} - \mathbf{t} $ PairRE: $- \mathbf{h} \circ \mathbf{r}^H - \mathbf{t} \circ \mathbf{r}^T $	MuRP: $-d_{\mathbb{H}}(\mathbf{x}_h^{(r)}, \mathbf{x}_t^{(r)})^2 + b_h + b_t$ ATTH: $-d_{\mathbb{H}}(\mathbf{Q}(\mathbf{h}, \mathbf{r}), \mathbf{t}^H)^2 + b_h + b_t$ HyperKA: $-d_{\mathbb{H}}(\mathbf{x}_h^{(0)} \oplus \mathbf{x}_r^{(0)}, \mathbf{x}_t^{(0)})$	MainfoldE: $- \varphi(\mathbf{h}) + \varphi(\mathbf{r}) - \varphi(\mathbf{t}) ^2$ MuRS: $-d_{\mathbb{S}}(\mathbf{x}_h^{(r)}, \mathbf{x}_t^{(r)})^2 + b_h + b_t$ SEA: $- e - \mathbf{q}_i + b_h + b_e$
Illustration			

in Cartesian coordinates. Moreover, QuatE [225] extends complex space to quaternion where there are two planes of rotations. Inspired by RotatE, Tang et al. [162] extend RotatE from 2D complex domain to high dimensional space with orthogonal transformations, which preserves the ability of modelling different patterns while achieving better performance. Furthermore, numerous RotatE-based models [46, 71, 88, 89, 183] exist, which expand upon the foundational principles of the original RotatE framework. For example, Rotate3D [46] and Rotate4D [89] define rotational relations in higher dimensional space. However, it remains challenging for KGE models to handle complex relations. To mitigate this problem, PairRE [25] uses paired vectors for each relation $\mathbf{r} = [\mathbf{r}^H, \mathbf{r}^T]$, where the value of \mathbf{r}^H and \mathbf{r}^T can be changed to fit the complex relations. Through the scoring function, relation vectors can project entities to arbitrary positions inside a unit circle lying on the Cartesian coordinates. Further analysis also proves that PairRE can capture subrelation in addition to those that RotatE can model [158].

Other Cartesian coordinate-based KGE models have also emerged recently. For example, TripleRE [219] inherits PairRE's projection part and builds the translation part by its own way. InterHT [176] merges the information from tail entity to head entity representation. TranS [229] is proposed to efficiently capture single relations and HousE [95] utilise Householder parameterisation [70] to capture crucial relation patterns. CompoundE [50] successfully embeds KGs by leveraging three fundamental Euclidean geometric operations. CompoundE3D [51] updates the compound transformation to further match the rich underlying characteristics of a KG. ConE [117] combines an explicit relation and a latent relation as collaborative relation for solving circular relation problems. The above geometric models based on Cartesian coordinates are illustrated in Figure 6. *In summary, the prevalent geometric transformations employed in KGE include translation, rotation, reflection, and scaling. These transformations have been proven effective in capturing essential relational patterns and mapping characteristics.*

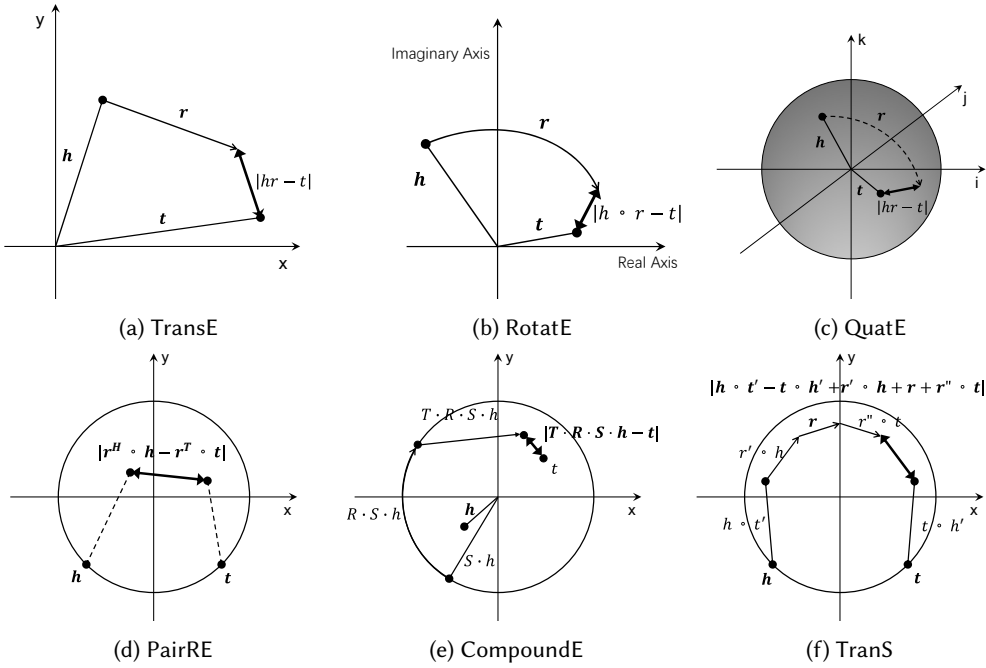


Fig. 6. KGE models in Cartesian Coordinate. “ \leftrightarrow ” illustrates the distance (scoring) function.

Polar Coordinate. In order to naturally reflect the hierarchy of KGs, HAKE [231] models entities and relations in the polar coordinate system. The radial coordinate aims to model entities ($\mathbf{h}_m, \mathbf{t}_m, \mathbf{r}_m \in \mathbb{R}^k$) at different levels, and the angular coordinate aims to distinguish entities ($\mathbf{h}_p, \mathbf{t}_p, \mathbf{r}_p \in [0, 2\pi)$) at the same level of hierarchy. Combining the modulus and phase information, HAKE significantly outperforms the SOTA hierarchy-based models while also completely capturing relational patterns. Although H²E [185]’s representation space belongs to hyperbolic geometric, it also uses modulus and phase information to embed entities— which is named Hyperbolic Polar Embedding. Modulus embedding models the inter-level hierarchy and Phase Embedding models the intra-level hierarchy. Recently, HBE [132] is also a hyperbolic geometry-based KGE model but with polar coordinate system by Mobius multiplication and Mobius addition in an extended Poincare ball.

Spherical Coordinate. STKE [184] represents entities and relations in a spherical coordinate system, but for temporal knowledge graphs. Each entity contains the radial part r , the azimuth part θ and the polar part φ . With the help of spherical coordinates, STKE treats temporal changes as scaling and rotation of entity embeddings, which can dynamically distinguish different time-constrained entities.

3.2.2 Hyperbolic Geometry

MuRP [4] is proposed to embed hierarchical multi-relational data in the Poincaré ball of hyperbolic space, where the Poincaré ball is defined as a d -dimensional manifold with the form: $\mathbb{B}_c^d = \{x \in \mathbb{R}^d : c \|x\|^2 < 1, c > 0\}$. Compared with Euclidean space, hyperbolic surface can be seen to have more spaces to represent entities and capture hierarchy information with increasing radius [147], so that MuRP outperforms Euclidean KGE models and achieves better performance, especially in

hierarchical datasets. Instead of learning in a hyperbolic space with fixed curvature as in MuRP, ATTH [24] leverages expressive hyperbolic isometries to simultaneously capture logical patterns and hierarchies. For each relation (e.g., rotation, reflection), it learns a specific absolute curvature c_r to avoid precision errors. Based on ATTH, models in different conditions are also proposed (i.e., ATTE, ROTTE/H, REFE/H). As also mentioned in section about polar coordinate, HBE [132] employs an extended Poincaré ball to capture hierarchical structures using polar coordinate system (Poincaré disk) to solve the boundary constraints which might happen in conventional Poincaré ball. Sun et al. [157] also represent KG embedding in a hyperbolic space by HyperKA, but firstly incorporate hyperbolic translation embedding with graph neural network (GNN). However, with the constant emergence of hyperbolic models, Kai Wang et al. [179] could not help asking: *Is Hyperbolic Geometry necessary?* Considering that hyperbolic-based model consistently necessitates greater computational complexity, which subsequently leads to the requirement for increased training resources, it raises the question of whether the benefits outweigh the associated costs? To tackle this issue, RotL and Rot2L are proposed to simplify the hyperbolic operation in RotH. By defining Flexible Addition, RotL can reduce the computation complexity of RotH [24] and save over 50% training time. However, it should not be ignored that knowledge graphs have multiple mixed relations, and excessive focus on hierarchical information often neglects capturing other information. Hyperbolic hierarchical transformations are introduced in HypHKGE [237] to extract hierarchies. MuRMP [186] utilises the mix-curvature model combined with GNN to better capture intrinsic heterogeneous structure in the KGs. UltraE [204] considers an ultrahyperbolic manifold to overcome the non-hierarchical embedding problems. Several contemporary KGE models (e.g., SEPA [55] and FFTAtH [199]) grounded in hyperbolic spaces have also demonstrated commendable performance.

3.2.3 Spherical Geometry

Before introducing the spherical geometry models, in order to avoid the confusion of the concepts of spherical coordinate system and spherical geometry, we first make a specific distinction between them. The spherical coordinate system mentioned above can be consider as a tool to describe points' positions naturally. And the basic concepts behind it are points and straight lines based on the Euclidean geometry. However, in spherical geometry [63], the basic concepts are point and great circles where any two lines meet in two points, and there are also no parallel lines. *Spheres are compatible with ring structures as the circular pattern of the vector field generated by spherical embedding has a natural circularity [160]. Consequently, spherical-based models also can yield competitive outcomes on complex relational datasets.*

TransC [116], in order to differentiate instances and concepts, encodes each concept in knowledge graph as a sphere (e.g. concept c_i is encoded as a sphere $s_i(\mathbf{p}_i, m_i)$, where \mathbf{p}_i denotes the centre, m_i denotes the radii) and each instance as a vector in the same semantic space. For example, for distinguishing the relations between concepts and sub-concepts (i.e. *subClassOf*), TransC construct several possible positions (e.g. inclusion, intersection, separation) between two concept spheres for different conditions. Comparing with TransC, HypersphereE [37] extends the sphere into hypersphere so as to not neglect the uncertainty of instances. Another special model named ManifoldE [198], which expands point-wised embedding to manifold-based embedding. Instead of adopting previous translational-based principle $\mathbf{h} + \mathbf{r} = \mathbf{t}$, ManifoldE employs manifold-based principle $\mathbf{M}(\mathbf{h}, \mathbf{r}, \mathbf{t}) = D_r^2$ (\mathbf{M} is the manifold function) in sphere and hyperplane respectively. In Sphere condition, entity which is always considered as a point in classic models is now extended to a whole high dimensional sphere. Through this mean, ManifoldE avoids much noise so as to best distinguish true facts. The smoothness of the sphere surface makes the embedding very flexible, and there can be countless mappings from the centre of the circle to the surface. As

we know that KGs usually contain rich types of structure such as hierarchical and cyclic typed structures. Embedding KGs in single curvature space, such as Euclidean or hyperbolic space, overlooks the intrinsic heterogeneous structures of KGs, and therefore cannot accurately capture their structures. Hence, M^2GNN , as a mixed-curvature model, is designed to address this issue. The scoring function is defined as $\phi_{\mathbb{P}_{K_o, K_h, K_s}}^{d_o, d_h, d_s}(e_h, r, e_t)$, where \mathbb{P} denotes the mixed-curvature space, d_o, d_h, d_s are the dimensions of the component spaces of Euclidean, hyperbolic, and spherical, $K_o/h/s$ are the corresponding curvature of spaces. Recently, SEA [55] utilises spherical geometry to consolidate various extant representations of KGE queries, thereby capturing diverse logical and structural patterns. Intriguingly, SEPA [55], an alternative version of the SEA, can also be projected onto the Poincaré ball to encompass more intricate structural representation.

3.3 Analytical Structure

An analytical structure is usually thought of as a structure of having a measure. For instance, metric (or distance) is well defined in Euclidean space so that we can integrate, differentiate and other analytical operations. Similarly, in the probability space, the probabilistic measure is defined, so it can consider as an analytical structure. In this section, we will deeply dig into the KGE models by dividing them into two main spaces: Probability Space and Euclidean Space.

3.3.1 Probability Space

To the best of our knowledge, KG2E [65] is the first density-based KGE model which represents each entity/relation by a multi-dimensional Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ in probability space, where the mean vector $\boldsymbol{\mu}$ indicates its position and the covariance metric Σ indicates the corresponding (un)certainly which impacts on others. In addition, two similar methods based on KL-divergence and expected likelihood, are proposed to inspect the difference of asymmetric and symmetric respectively. Previous models have formally considered the issue of multiple relation semantics in KGs. However, the traditional translational-based models always assign only one vector for one vector, ignoring the fact that a relation may have multiple meanings. Thus, TransG [197] is proposed by leveraging a Bayesian non-parametric infinite mixture model to handle multiple relation semantics by generating multiple translation components for a relation. In TransG, entities are generated by a certain Gaussian distribution, where the m -th component translation vector of relation r is represented as: $\boldsymbol{\mu}_{r,m} = \mathbf{t} - \mathbf{h} \sim \mathcal{N}(\boldsymbol{\mu}_t - \boldsymbol{\mu}_h, (\sigma_h^2 + \sigma_t^2)E)$. Through this process, TransG would automatically select the best match between h , t and r . Other probabilistic-based models such as DBKGE [106] and GaussianPath [175] also harness the uncertainty of KGs by Gaussian representation. DirE [177] is proposed by embedding entities as Dirichlet distributions and relations as multinomial distributions. This method uses Bayesian inference to assess the relationships between entities and subsequently learns binary embeddings of knowledge graphs for modelling intricate relation patterns and uncertainty. Recently, ItôE [124] formulates the relations in a KG as stochastic Itô processes, enabling transitions between two nodes to occur with an associated likelihood. This approach permits ItôE to represent multiple stochastic trajectories including loops connected to paths and is mathematically substantiated as a generalisation of several state-of-the-art models. The aforementioned methodologies demonstrate that *probabilistic embedding is not only capable of acquiring unstructured patterns but also adept at capturing additional uncertain information* [208].

3.3.2 Euclidean Space

In order to improve the structure preservation capabilities of KGE models, Nayyeri et al. [125] propose a novel KGE model named FieldE which employs ordinary differential equations (ODEs) for embedding KGs into a Euclidean space. Each entity is represented by a vector in \mathbb{R}^n denoted by $\mathbf{e}(t)$

and each relation is represented as a vector field f_{θ_r} on a Riemannian Manifold, where $\mathbf{e}(t)$ lies on a trajectory (continuous) on the manifold \mathcal{M} solving the ODE: $\frac{d\mathbf{e}(t)}{dt} = f_{\theta_r}(\mathbf{e}(t))$. Hence, FieldE could capture the continuity of changes in the embedding space and describe the underlying geometry by nature. ODE method can also be used in temporal KGs, in which TANGO [61] is proposed to learn continuous-time representations of entities and relations dynamically by Neural ODE method [26]. *These analytical methods facilitate the acquisition of dynamic and continuous representations of entities and relations, which in turn enhance memory efficiency, adaptive computation, and parameter effectiveness in various subsequent tasks [84].* In addition to continuous analysis, other analytical perspectives such as derivability, differentiability, and integrability are also worth exploring in KGEs.

4 Downstream Tasks of Knowledge Graph Embedding

After introducing a systematic review of existing KGE models from the perspective of mathematical structure, this section focuses on KGE-based downstream tasks. We highlight some important and popular applications which are usually employed to evaluate the performance of embedding models. After summarising and comparing the performance of some KGE models, we examine their strengths and weaknesses from diverse spatial perspectives and offer some advice for building KGE models. The summary of the advantages and disadvantages of different KGE models from the spatial perspective is provided in Table 9.

In what follows, we first describe the process of **Link prediction**, a fundamental task in KGE domain, and focus on one popular task: **Hierarchy Acquisition** in the link prediction scenario. In addition, we analyse and discuss the task of **Pattern Inference**. Additionally, we present the model's results regarding the time/space complexity and discuss the applications of the KGE models in other downstream tasks. We also provide some suggestions for building KGE models based on the empirical results in Section 4.6.

4.1 Link Prediction

Link prediction aims to predict the existence of edges (triples) in knowledge graphs, which is a fundamental task since many existing KGs have missing facts or incorrect edges [128]. In particular, the task of link prediction is often formulated as predicting missing entity in an incomplete fact triple, i.e., predicting head entity h in $(?, r, t)$, or tail entity t in $(h, r, ?)$, where $(?, r, t)$ and $(h, r, ?)$ denote fact triples with missing entities. For example, given a triple $\langle ?, subclass, COVID-19 \rangle$, the goal is to predict the superclass of COVID-19. Therefore, link prediction is also referred to as knowledge graph completion [128], entity prediction [108] or entity ranking [12].

Given learned entity and relation embeddings with KGE methods, link prediction is carried out through a ranking procedure. Specifically, in order to predict the tail entity of an incomplete fact triple $(h, r, ?)$, we take each entity t' in the KG as a candidate answer and calculate the plausibility of the triple (h, r, t') , which is achieved by calculating the score function of the employed KGE method. For example, for KGE methods that learn KG embeddings in Euclidean space, the translation-based score function $s(\mathbf{h}, \mathbf{r}, \mathbf{t}) = -\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|_{1/2}$ is often used to assign scores for triples. In contrast, for KGE methods that learn embeddings in complex space, the rotation-based score function $s(\mathbf{h}, \mathbf{r}, \mathbf{t}) = -\|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\|^2$ is often utilised. After obtaining the scores of candidate answers, we can rank those candidate entities in the descending order of their scores, and select the highest ranked entity as the prediction result. A similar procedure can also be used to predict the missing head entity h in $(?, r, t)$. For evaluation, a common practice is to record the ranks of correct answers in the previous ranked list and leverage those ranks to calculate the evaluation metrics.

Table 4. The link prediction results on the WN18RR and FB15k-237 datasets, which are classified by its geometric space. The best scores of 32-dimensional models are in **bold**, the second best scores are underlined and the best average scores are **coloured**. TransE, ComplEx, QuatE, RotatE, MuRE, MuRMP, and 5★E results are taken from [204]. TuckER, RefE, RotE, AttE, MuRP, RefH, RotH, AttH, and Rot2L results are derived from [179]. MuRMP is a mix-curvature model. UltraE is an ultrahyperbolic-based model, in which its best results (when $q = 6$) are chosen. Other results are taken from their original paper. * denotes: DFieldE_S and MuRS results are not obtained in the case of low dimensions ($d = 32$), but in high dimensions. Hence the average score of the spherical-based model is not comparable. [∇] denotes: GIE is a geometric interactive model and its result is also in high dimensional conditions, which is not added to the average calculation.

Type	Method	Year	FB15K-237			WN18RR		
			MRR	Hits@10	Hits@1	MRR	Hits@10	Hits@1
Euclidean-based Models	TransE [14]	2013	0.295	0.466	0.210	0.366	0.515	0.274
	ComplEx [168]	2016	0.287	0.456	0.203	0.421	0.476	0.391
	QuatE [225]	2019	0.293	0.460	0.212	0.421	0.467	0.396
	RotatE [158]	2019	0.290	0.458	0.208	0.387	0.491	0.330
	TuckER [5]	2019	0.306	0.475	0.223	0.428	0.474	0.401
	MuRE [4]	2019	0.313	0.489	0.226	0.458	0.525	0.421
	HAKE [231]	2020	0.296	0.463	0.212	0.416	0.467	0.389
	RefE [24]	2020	0.302	0.474	0.216	0.455	0.521	0.419
	RotE [24]	2020	0.307	0.482	0.220	0.463	0.529	0.426
	AttE [24]	2020	0.311	0.488	0.223	0.456	0.526	0.419
	Rot2L [179]	2021	0.326	0.503	0.237	0.475	0.554	0.434
	EucHKGE [237]	2021	0.319	0.499	0.228	0.462	0.534	0.425
ItδE _R [124]	2023	0.330	0.508	0.242	0.455	0.548	0.404	
<i>Avg_score</i>	-	0.306	0.479	0.220	0.436	0.510	0.395	
Hyperbolic-based Models	MuRP [4]	2019	0.323	0.501	0.235	0.465	0.544	0.420
	RefH [24]	2020	0.312	0.489	0.224	0.447	0.518	0.408
	RotH [24]	2020	0.314	0.497	0.223	0.472	0.553	0.428
	AttH [24]	2020	0.324	0.501	0.236	0.466	0.551	0.419
	HypHKGE [237]	2021	0.330	0.510	0.240	0.475	0.556	0.432
	DFieldE _P [125]	2021	0.330	0.510	0.250	0.480	<u>0.570</u>	<u>0.440</u>
	FFTAttH [199]	2022	0.331	0.517	0.239	0.476	0.558	0.432
	ItδE _P [124]	2023	-	-	-	0.474	0.574	0.426
SEPA [55]	2023	0.332	0.509	0.243	<u>0.481</u>	0.562	0.441	
<i>Avg_score</i>	-	0.325	0.504	0.236	0.471	0.554	0.427	
Spherical-based Models	DFieldE _S [125]	2021	0.360	0.550	0.270	-	-	-
	MuRS [186]	2021	0.338	0.525	0.249	0.454	0.550	0.432
	ItδE _S [124]	2023	<u>0.334</u>	0.511	0.245	-	-	-
	<i>Avg_score</i> *	-	0.344	0.529	0.255	0.454	0.550	0.432
Mixed Models	MuRMP [186]	2021	0.319	0.502	0.232	0.470	0.547	0.426
	UltraE [204]	2022	0.338	<u>0.514</u>	<u>0.247</u>	0.483	0.555	0.425
	GIE [∇] [21]	2022	0.362	0.552	0.271	0.491	0.575	0.452
	<i>Avg_score</i>	-	0.329	0.508	0.240	0.477	0.551	0.426

4.2 Hierarchy Acquisition

Nowadays, more and more KGE models not only aim to obtain SOTA in link prediction task, but also pay special attention to whether they can capture hierarchy properties. One main reason why these works focus on capturing hierarchy structures is that: **Multi-relational knowledge graphs often exhibit multiple simultaneous hierarchies** [33, 165]. However, conventional models

(e.g., TransE, RotatE) merely emphasis on capturing these hierarchies. Therefore, we focus on KGE models from the view of mathematical space, aiming to find the most appropriate space to capture multi-layered information. In this part, we first introduce what hierarchy exactly is, and then draw our conclusion by comparing the performance of KGE models in different spaces on hierarchy-contained datasets.

What is Hierarchy? Semantic hierarchy is a ubiquitous property in knowledge graphs. Some relations can induce various hierarchical structure. For instance, *chair* is at a higher lever than *armchair*, *fighting_chair* under the relation *hypernym*, and *armchair*, *fighting_chair* are parent nodes to their part: *backrest*, *leg* with the relation *has_part*. Such hierarchies can be treated as "tree-like" structures intuitively.

Results. We summarise the performance of KGE models in link prediction task on two hierarchical datasets: WN18RR [33] and FB15K-237 [165], where the curvature metric ξ_G (The lower the metric ξ_G is, the more hierarchical the knowledge graph is [57].) of two datasets are -2.54 and -0.65, respectively. We summarise two key findings: **(1). Non-Euclidean (e.g., hyperbolic-based, spherical-based) models typically demonstrate a better capacity to capture various KG structures in low dimensions, as opposed to their Euclidean counterparts. (2). Within high-dimensional conditions, both non-Euclidean and Euclidean models exhibit comparable performance in representing KGs.**

To ensure a fair comparison, we substantiate our conclusions through multiple aspects. Firstly, we summarise the link prediction results of some state-of-the-art KGE models on the WN18RR and FB15k-237 datasets in Table 4. By analysing the empirical results of these models in Table 4, we can preliminarily infer that, by measured by the average scores metric (i.e., the *Avg_score*), Euclidean spatial models generally exhibit inferior performance compared to non-Euclidean spatial models in low-dimensional embeddings. In particular, hyperbolic-based models significantly outperform Euclidean baselines on WN18RR and FB15K-237 (highlighted in orange). Subsequently, Table 5 enumerates models based on distinct spaces from the same article to corroborate the conclusion. For instance, the performance of RefH/RotH/AttH is evidently superior to their Euclidean counterparts, RefE/RotE/AttE, across multiple evaluation metrics. Concurrently, Tables 5 and 6 present KGE models over the same time period, effectively eliminating the influence of temporal factors on the analysis. Therefore, **(1).** In low dimensions, hyperbolic embeddings present superior performances compared to Euclidean-based embeddings. Concurrently, spherical embedding also yields favourable results (See Table 4, 5 and 6). This phenomenon can be mathematically elucidated that: spheres are congruent with ring structures (as depicted in Figure 1b) due to the circular nature of the vector field produced by spherical embedding exhibits circularity, facilitating the capture of cyclic structures [160]. In contrast, vector fields in hyperbolic spaces consistently operate from the narrower regions of the manifold and progress towards its broader sides [125]. The curvature of hyperbolic spaces in low-dimensional settings exhibits a direct correlation with the calculated graph curvature [24], making it suitable for tree-like or hierarchical configurations [45, 126, 127, 163]. **(2).** However, when the embedding dimension is large, Euclidean, hyperbolic and spherical embedding methods perform similarly across all datasets (See Table 5 and 6). We explain this behaviour by noting that, given sufficiently large dimensions, both Euclidean and non-Euclidean spaces possess ample capacity to represent intricate hierarchies present in KGs. Consequently, the disparity among manifold selections in high-dimensional settings appears to be minimal.

Therefore, hyperbolic space provides effective methods for studying low-dimensional embeddings while maintaining underlying hierarchical structures, allowing hyperbolic-based models to embed tree-like structures with minimal distortion in merely two dimensions [24]. And the spherical space demonstrates the capacity to encapsulate ring structures, owing to its inherent property of

Table 5. The link prediction results in different mathematical spaces from the **same paper** for low-dimensional ($d = 32$) and high-dimensional (best for $d \in \{200, 400, 500\}$) embeddings. Models from the same article in different spaces are identified by identical symbols, whereas models originating from separate articles are distinguished by different symbols. [♣]: MuRE/P are from [4]; [♠]: Results are from [4], where ♠; denotes models using different geometric transformation; [◇]: Euc/HypHKGE are from [237]; [♥]: ItôE_{R/P/S} are from [237]; [‡]: MuRS/MP are from [186]; The best scores are in **bold**, the second best scores are underlined.

Space	Method	WN18RR						FB15K-237					
		low-dimension			high-dimension			low-dimension			high-dimension		
		MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10
E	MuRE ♣ (2019)	0.458	0.421	0.525	0.475	0.436	0.554	0.313	0.226	0.489	0.336	0.245	0.521
	RefE ♠ ₁ (2020)	0.455	0.419	0.521	0.473	0.430	0.561	0.302	0.216	0.474	0.351	0.256	0.541
	RotE ♠ ₂ (2020)	0.463	0.426	0.529	<u>0.494</u>	0.446	<u>0.585</u>	0.307	0.220	0.482	0.346	0.251	0.538
	AttE ♠ ₃ (2020)	0.456	0.419	0.526	0.490	0.443	0.581	0.311	0.223	0.488	0.351	0.255	0.543
	EucHKGE ◇ (2021)	0.462	0.425	0.474	0.493	0.447	0.583	0.319	0.228	0.499	<u>0.354</u>	<u>0.261</u>	<u>0.545</u>
ItôE _R ♥ (2023)	0.455	0.404	0.548	-	-	-	<u>0.330</u>	0.242	0.508	-	-	-	
H	MuRP ♠ (2019)	0.465	0.420	0.544	0.481	0.440	0.566	0.323	0.235	0.501	0.335	0.243	0.518
	RefH ♠ ₁ (2020)	0.447	0.408	0.518	0.461	0.404	0.568	0.312	0.224	0.489	0.346	0.252	0.536
	RotH ♠ ₂ (2020)	0.472	0.428	0.553	0.496	0.449	0.586	0.314	0.223	0.497	0.344	0.246	0.535
	AttH ♠ ₃ (2020)	0.466	0.419	0.551	0.486	0.443	0.573	0.324	0.236	0.501	0.348	0.252	0.540
	HypHKGE ◇ (2021)	0.475	0.432	0.556	0.494	<u>0.448</u>	0.584	0.330	<u>0.240</u>	<u>0.510</u>	0.351	0.258	0.541
ItôE _P ♥ (2023)	<u>0.474</u>	<u>0.426</u>	0.574	-	-	-	-	-	-	-	-	-	
S	MuRS † (2021)	-	-	-	0.454	0.432	0.550	-	-	-	0.338	0.249	0.525
	ItôE _S ♥ (2023)	-	-	-	-	-	-	0.334	0.245	0.511	-	-	-
M	MuRMP † (2021)	0.470	0.426	0.547	0.481	0.441	0.569	0.319	0.232	0.502	0.358	0.273	0.561

Table 6. The scores or average scores of KGE models in Euclidean space (E) and Hyperbolic space (H) during the **same period**. [2019] are the scores of MuRE (E) and MuRP (P); [2020] are the average scores of Ref/Rot/AttE (E) and Ref/Rot/AttH (H); [2021] are the scores of EucHKGE (E) and HypHKGE (H).

Year	Space	WN18RR						FB15K-237					
		low-dimension			high-dimension			low-dimension			high-dimension		
		MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10
2019	E	0.458	0.421	0.525	0.475	0.436	0.554	0.313	0.226	0.489	0.336	0.245	0.521
	H	0.465	0.420	0.544	0.481	0.440	0.566	0.323	0.235	0.501	0.335	0.243	0.518
2020	E	0.458	0.421	0.527	0.485	0.444	0.575	0.304	0.219	0.481	0.348	0.254	0.540
	H	0.461	0.418	0.540	0.481	0.432	0.575	0.322	0.227	0.495	0.346	0.250	0.537
2021	E	0.462	0.425	0.474	0.493	0.447	0.583	0.319	0.228	0.499	0.354	0.261	0.545
	H	0.475	0.432	0.556	0.494	0.448	0.584	0.330	0.240	0.510	0.351	0.258	0.541

extracting circularity. Nevertheless, this does not mean that non-Euclidean models are necessarily superior to other spaces since some Euclidean-based models can also achieve high performance through ingenious tricks, such as ReflectE [222], CompoundE [50]. In addition, the results of the mixed models (coloured in purple) which learn embeddings in multiple spaces are more outstanding, which means that better performance can be achieved by using multiple geometries simultaneously with a suitable hybrid method.

4.3 Patterns inference

Another popular task is about exploring the pattern inference capability since large-scale KGs always exhibit various types of relationships. As described in [13], a excellent model should be able to learn all combinations of these properties: (a) **symmetry** (e.g., marriage, is_similar_to). (b) **antisymmetry** (e.g., father_of). (c) **inversion** (e.g., hypernym and hyponym). (d) **composition** (e.g., my mother's father is my grandpa.). However, looking back at the development of previous KGE models, it was a difficult process to build a model that could capture all of the above attributes simultaneously. In this section, we will mainly analyse the advantages of models focusing on inferring relational patterns, and summarise the key factors of capturing those properties.

First, we give these four important patterns' formal definitions as follows [158]:

Definition 4.1. A relation r is **symmetric** if $\forall x, y$

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

A clause with such form is a **symmetry** pattern.

Definition 4.2. A relation r is **antisymmetric** if $\forall x, y$

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

A clause with such form is a **antisymmetry** pattern.

Definition 4.3. Relation r_1 is **inverse** to relation r_2 if $\forall x, y$

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

A clause with such form is a **inversion** pattern.

Definition 4.4. Relation r_1 is **composed** of relation r_2 and relation r_3 if $\forall x, y, z$

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

A clause with such form is a **composition** pattern.

Next, we analyse specific operations in different spaces to explain how existing models can infer and model those patterns. Table. 7 shows the patterns inference capabilities of some main KGE works, and their corresponding spaces and operators.

Addition. The KGE models with addition operation as their core are often found among distance-based models such as SE [15], TransE [14] and TransE's variants. The characteristic of addition is that it is easy to establish the connection between vectors, and the complexity is very low. In SE [15], the scoring function is too sketchy to capture any of these patterns; By defining $h + r \approx t$, TransE [14] makes good use of vector addition to establish the relationship between relations and entities, and can capture three patterns except symmetry. Since then, a large number of translation-based models (e.g., TransH [190], TransR [109], etc.) still retain addition operation but carry out special operation $f_r(\cdot)$ (mostly matrix multiplication based on r). What remains unchanged is that the core of operation is still Addition, that is, to a large extent, addition corresponds to the concept of "translation". However, with the added projection parameters, these models are unable to encode inverse and composition. Although they have made progress in dealing with complex relations, the Trans's variants are a step back in terms of modelling patterns — a drawback of the simplicity of addition.

Product. The term "product" refers to the results of one or more multiplications. Most existing SOTA models use product operation, such as ComplEx [168] and RotatE [158]. Products in different spaces have different properties to help the KGE models infer more patterns. Here we mainly divide the product operations into the following groups and analyse them in detail.

- *Dot Product.* The dot product may be defined algebraically or geometrically. With algebraic definition, inner product is a way to multiply vectors together, with the result of this multiplication being a scalar. With geometric definition, the notions of length and angles can be defined by means of the dot product. In KGE, the fundamental purpose of product is to expect to establish complex relations between relations and entities through multiplication rather than simple addition. For example, the scoring function in DistMult [209] is defined as $f_r(h, t) = \mathbf{h}^T \text{diag}(\mathbf{r}) \mathbf{t} = \sum_i [\mathbf{r}]_i \cdot [\mathbf{h}]_i \cdot [\mathbf{t}]_i$, we can see that each score in the summation is

a direct multiplication of $\mathbf{h}_i, \mathbf{r}_i, \mathbf{t}_i$; similar in ComplEx [168], the scoring function is extended to complex space. Note that they are both RESCAL [130]'s extensions, and RESCAL was originally built for implicit semantic matching by factorisation, which is one of the latent advantages of dot products.

- *Hadamard Product*. Hadamard Product (also known as element-wise product) is a binary operation by which the elements corresponding to the same row and columns of given vectors/matrices are multiplied together to form a new vector/matrix. The difference between the dot product and the Hadamard product operationally is the aggregation by summation. The dot product of two vectors gives only a scalar number while the Hadamard product of two vectors gives a complete vector, which preserves a large amount of transformed information of KGs. As in RotatE [158], through the principle $\mathbf{t} = \mathbf{h} \circ \mathbf{r}$, \mathbf{r} becomes an element-wise rotation from the head entity to the tail entity; In HAKE [231], each \mathbf{r}_i is regarded as a scaling transformation between two moduli; Cross interaction operations are also applied by utilising Hadamard product in CrossE [227]. Similarly uses of Hadamard product also appear in ComplEx [168], PairRE [25], etc. In general, the Hadamard product is closely related to the concept of “rotation”. Besides, the Hadamard product appears in lossy compression algorithms such as JPEG, can also be used for describing NN as LSTM [66], GRU [29] or enhancing, suppressing or masking image regions.
- *Other Products*. Other subsequent product operations are mostly extended based on the existing advantages of Dot/Hadamard product. Zhang et al. [225] believed that latent inter-dependencies between all components are aptly captured with Hamilton product, encouraging a more compact interaction between entities and relations; What's more advanced than RotatE is that DualE [20] adopts quaternion inner product operation to model both translation and rotation, which improves the capability of inferring three important patterns.

Other Operations. Other operations are mostly those with exclusive rules under some special conditions, but show strengths in some specific tasks (e.g., multi-relation tasks). For example, simple matching in SME [12]/RESCAL [130], although it only uses simple linear/bilinear matching, it opens a new field for semantic matching KGE model; Circular correlation in HoLE [129], which makes compressions of pairwise tensor product to enhance its efficiency; Orthogonal Transform [129] that aims to Unleash the original potential of RotatE into higher dimensions; Mobius addition and attention in ATTH [24] are utilised to represent relations as parameterised geometric operations that directly map to logical properties.

4.4 Knowledge Infusion To Enhance Other Domain Applications

Apart from the aforementioned applications that are appropriate for KG embeddings, there are other wider fields where KG representation learning could be infused to enhance. Knowledge graph based questions answering (KGQA) is a fundamental, but still challenging task. It recognises the user's question input in order to obtain the accurate answers composed of KG entities. Existing methods include semantic parsing models [86, 100, 216], reinforcement learning [139, 205], and so on. Knowledge reasoning is a process of using known knowledge to infer new knowledge. Researchers always use machine learning methods [188, 189, 232] to infer potential relations between entity pairs and identify erroneous knowledge based on existing data automatically. There are lots of other external applications based on knowledge graph embedding are still worth exploring, such as Recommendation System [9, 178, 236], Information Retrieval [40, 111, 141] and other specific domain (e.g. Cyber Security [69, 99], Biomedicine [22, 172], etc.). The recent surge of LLMs [17, 19, 30, 31, 149, 166]) has demonstrated exceptional proficiency in handling diverse Natural Language Processing (NLP) tasks, including question answering, machine translation, and

Table 7. The pattern inference abilities of several models by using different operators in certain spaces. SE, TransE, TransR, DisMult, ComplEx, and RotatE results are taken from [158]. Other results come from origin papers. \mathbb{R} represents the Euclidean space. \mathbb{R}_p represents the Euclidean space with polar coordinate system [231]. \mathbb{C} represents the complex vector space. \mathbb{G} represents the group. \mathbb{H} represents the hyperbolic space. \mathbb{UH} represents the ultra-hyperbolic space. \mathbb{Q} represents the quaternion vector space. \mathbb{BQ} and \mathbb{DQ} represents the biquaternion vector space and dual quaternion vector space, respectively, both of them are special cases of \mathbb{Q} .

Method	Manifold	Operator	Relation Patterns			
			<i>Sym</i>	<i>Asym</i>	<i>Inv</i>	<i>Comp</i>
SE [15]	\mathbb{R}	Addition(+)	\times	\times	\times	\times
TransE [14]	\mathbb{R}	Addition(+)	\times	\checkmark	\checkmark	\checkmark
TransR [109]	\mathbb{R}	Addition(+)	\checkmark	\checkmark	\times	\times
DistMult [209]	\mathbb{R}	Inner Product($\langle \cdot \rangle$)	\checkmark	\times	\times	\times
HolE [129]	\mathbb{R}	Circular Coorelation(\star)	\checkmark	\checkmark	\checkmark	\times
HAKE [231]	\mathbb{R}_p	Hadamard Product(\odot)	\checkmark	\checkmark	\checkmark	\checkmark
PairRE [25]	\mathbb{R}	Hadamard Product(\odot)	\checkmark	\checkmark	\checkmark	\checkmark
ComplEx [168]	\mathbb{C}	Hadamard Product(\odot)	\checkmark	\checkmark	\checkmark	\times
RotatE [158]	\mathbb{C}	Hadamard Product(\odot)	\checkmark	\checkmark	\checkmark	\checkmark
ExpressivE [133]	\mathbb{R}	Hadamard Product(\odot)	\checkmark	\checkmark	\checkmark	\checkmark
OTE [162]	\mathbb{R}	Orthogonal Transform(ϕ)	\checkmark	\checkmark	\checkmark	\checkmark
QuatE [225]	\mathbb{Q}	Hamilton Product(\otimes)	\checkmark	\checkmark	\checkmark	\times
BiQUE [59]	\mathbb{BQ}	Hamilton Product(\otimes)	\checkmark	\checkmark	\checkmark	\checkmark
DualE [20]	\mathbb{DQ}	Dual Quaternion Product($\langle \otimes \rangle$)	\checkmark	\checkmark	\checkmark	\checkmark
Rotate3D [46]	\mathbb{R}	Rotational Product(\odot)	\checkmark	\checkmark	\checkmark	\checkmark
HousE [95]	\mathbb{R}	Householder Rotation(\mathcal{S})	\checkmark	\checkmark	\checkmark	\checkmark
ATTH [24]	\mathbb{H}	Möbius addition(\oplus^c)	\checkmark	\checkmark	\checkmark	\checkmark
DihEdral [206]	\mathbb{G}	Matrix Product(\odot^*)	\checkmark	\checkmark	\checkmark	\checkmark
UltraE [204]	\mathbb{UH}	Ultrahyperbolic Transform(f_r)	\checkmark	\checkmark	\checkmark	\checkmark

text generation. Several studies [28, 143, 159, 214, 232, 233] have been conducted to substantiate the benefits provided by models integrating knowledge graphs and large language models.

4.5 Model Complexity

In this section, we incorporate Table 8 to examine the time and space complexity of some KGE models. Based on the analyses in Table 8, we can draw the following conclusions. First, models which represent entities and relations as vectors (e.g., TransE, TransH, ComplEx, and UltraE) are more efficient. They usually have space and time complexity that scales linearly with entity dimension d . HolE needs more time complexity as it computes circular correlation via the discrete Fourier transform. Second, models which represent relations as matrices (e.g., TransR, SE, and RESCAL) usually have higher complexity in both space and time. Third, advanced methods (e.g., RotE/H, Rot2L, and ItôE) may lead to a linear increase in space complexity but not to order. RotE/H requires more relation parameters since relation transformation vectors and the learnable curvature for different relations are needed. It is important to mention that non-Euclidean models exhibit only marginal deviations from Euclidean models in terms of time or space complexity. Regarding the training cost, it has been demonstrated that hyperbolic embedding typically necessitates more training time than Euclidean embedding, owing to the fact that the Möbius operations in hyperbolic

Table 8. Comparison of state-of-the-art knowledge graph embedding models, along with their publisher, math space, time complexity, and space complexity. The complexity results are taken from [122, 181] or referenced from their corresponding papers. RotE/H results are obtained from Rot2L [179]. d and k are the embedding dimensionality of entities and relations, respectively (usually $d = k$). N_e and N_r are the numbers of entities and relations. θ denotes the average sparseness degree of projection matrices in TransSparse [76].

Method	Publisher	Math Space	Time Complexity	Space Complexity
SE [15]	AAAI 2011	Euclidean	$\mathcal{O}(d^2)$	$\mathcal{O}(N_e d + N_r d^2)$
RESCAL [130]	ICML 2011	Euclidean	$\mathcal{O}(d^2)$	$\mathcal{O}(N_e d + N_r d^2)$
TransE [14]	NeurIPS 2013	Euclidean	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + N_r d)$
TransH [190]	AAAI 2014	Euclidean	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + N_r d)$
TransR [109]	AAAI 2015	Euclidean	$\mathcal{O}(dk)$	$\mathcal{O}(N_e d + N_r dk)$
TransD [75]	ACL-IJCNLP 2015	Euclidean	$\mathcal{O}(\max(d, k))$	$\mathcal{O}(N_e d + N_r k)$
DistMult [209]	ICLR 2015	Euclidean	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + N_r d)$
ComplEx [168]	ICML 2016	Euclidean	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + N_r d)$
TransSparse [76]	AAAI 2016	Euclidean	$\mathcal{O}(dk)$	$\mathcal{O}(N_e d + (1 - \theta)N_r dk)$
HoLE [129]	AAAI 2016	Euclidean	$\mathcal{O}(d \log d)$	$\mathcal{O}(N_e d + N_r d)$
ANALOGY [110]	ICML 2017	Euclidean	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + N_r d)$
ConvE [33]	AAAI 2018	Euclidean	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + N_r d)$
RotE [24]	ACL 2020	Euclidean	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + 2N_r d)$
Rot2L [179]	ACL Findings 2021	Euclidean	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + 2(N_r + 5)d)$
It δ E $_{\mathbb{R}}$ [124]	ACL Findings 2023	Euclidean	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + N_r k)$
RotH [24]	ACL 2020	Hyperbolic	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + 3(N_r + 1)d)$
It δ E $_{\mathbb{P}}$ [124]	ACL Findings 2023	Hyperbolic	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + N_r k)$
ManifoldE $_{\mathbb{S}}$ [198]	IJCAI 2016	Spherical	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + N_r d)$
It δ E $_{\mathbb{S}}$ [124]	ACL Findings 2023	Spherical	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + N_r k)$
UltraE [204]	KDD 2022	Ultrahyperbolic	$\mathcal{O}(d)$	$\mathcal{O}(N_e d + N_r d)$

are far more complex than the Euclidean operations [179]. Simultaneously, the training cost of various models within the same space tends to fluctuate based on factors such as model size, mapping method [95] (linear or nonlinear), and the usage of acceleration algorithms [97, 180, 226]. Consequently, we conclude that determining the complexity of a KGE method is a multidimensional and comprehensive challenge. It is worth emphasising that our survey's primary objective is to analyse KGE methods from the standpoint of representation space, and as such, we do not further explore the intricacies of complexity.

4.6 Suggestions

In Section 4, we present the performance of different KGE models on different downstream tasks and applications from the mathematics spatial perspective. Here we summarise our analyses and provide suggestions and guidance for constructing KGE models: (1). Firstly, we analyse the performances of KGE models in different geometric spaces based on the empirical results in Table 4. The results show that the *hyperbolic-based* models have relatively better performance on the FB15K237 and WN18RR datasets. This is due to the fact that most existing KG datasets are known to have tree-like or hierarchical structures and thus favour hyperbolic embeddings [45, 126, 163]. Therefore, it is suggested to employ hyperbolic-based models for handling datasets (not limited to knowledge graphs) that exhibit hierarchical structure. Meanwhile, models that blend various geometric properties (e.g., UltraE [204], DGS [73], GIE [21] and Concept2Box [72]) have also achieved promising performance. Nonetheless, it is worth noting that employing sophisticated geometric embeddings (e.g., hyperbolic embedding) will lead to increased computational complexity,

Table 9. The advantages and disadvantages of different KGE models from the spatial perspective.

Perspective	Subclass/Subspace	Examples	Properties	Advantages	Disadvantages
Geometric Structure	Euclidean Geometry	TransE [14]; RotatE [158]; PairRE [25]; TripleRE [219]; TransS [229]; HopfE [8]; CompoundE [50]; HAKE [231]; H ² E [185]	The prevalent Euclidean geometric transformations can effectively capture simple relational patterns and characteristics.	<ul style="list-style-type: none"> It is easy and intuitive to bridge relationships between entities through a variety of geometric transformations, such as <i>translation</i>, <i>rotation</i>, <i>reflection</i> and <i>scaling</i>. 	<ul style="list-style-type: none"> Advanced geometric embedding methods (e.g., hyperbolic embedding) tend to be more complex than Euclidean embedding and also require more training time and cost [179].
	Hyperbolic Geometry	MuRP [4]; ATTH [24]; HBE [132]; HyperKA [157]; UltraE [204]; ItôE [124]; HypHKE [237]; H ² E [185]; H ² E [185]; HBE [132]	Hyperbolic surface has more (tree-like) spaces to represent and capture hierarchy information.	<ul style="list-style-type: none"> The distinctive and heterogeneous architecture of knowledge graphs is closely associated with geometric embedding (e.g., Ring KGs vs. Spheres [160]). 	<ul style="list-style-type: none"> Certain geometries might exhibit diminished benefits for graphs containing long paths [147].
	Spherical Geometry	ManifoldE [198]; GIE [21]; TransC [116]; MuRS [186]; HyperspherE [37]; SEA [55]	Sphere excels at capturing ring structures by its circularity nature of spherical embedding.	<ul style="list-style-type: none"> Non-Euclidean models are suitable for complex KGs tasks such as link prediction. 	
Algebraic Structure	Vector Space	TransE [14]; RESCAL [130]; TransH [190]; DisMult [209]; ComplEx [168]; RotatE [158]; QuatE [225]; DualE [20]; ConvE [33]; R-GCN [150]; KG-BERT [212]; DKRL [200]	With simple and efficient operations in vector space, the plausibility of facts can be easily measured by matching the latent semantics of entities and relations.	<ul style="list-style-type: none"> Using proper algebraic operations can help KGE models capture more important patterns of knowledge graphs (e.g., <i>symmetry</i>, <i>antisymmetry</i>, <i>inversion</i>, and <i>composition</i>). 	<ul style="list-style-type: none"> It should be noted that the demonstrations of various algebraic structures are solely dependent on the specific KG task itself, and the extent to which they can be generalised to other domains remains uncertain.
	Group	TorusE [38]; DihEdral [206]; NagE [211]; ModulE [23]; KGLG [39]; DensE [115]; GrpKG [210]	The definition of group can naturally satisfy the basic properties (e.g., <i>inversion</i> , <i>composition</i>) of KGs.	<ul style="list-style-type: none"> The plausibility of facts can be measured by matching latent semantics of entities and relations through efficient linear operations. [82, 129] 	<ul style="list-style-type: none"> Some operations, such as Möbius addition [27], may be less intuitive and consequently challenging to extend to numerous scenarios.
	Ring	MöbiusE [27]	Ring structure is helpful for orientation-related tasks due to its non-oriented surface.		
Analytic Structure	Probability Space	TransG [197]; KG2E [65]; DirIE [177]; ItôE [124]; GaussianPath [175]	Probabilistic embedding is capable of acquiring unstructured patterns and capturing uncertain information.	<ul style="list-style-type: none"> Concentrating on analytical properties (e.g., <i>uncertainty</i>, <i>continuity</i>, and <i>differentiability</i>) in the modelling process enables the KGE systems to capture stable and robust representations [124, 125]. 	<ul style="list-style-type: none"> The analytical nature primarily serves as an auxiliary component in the KGE modelling process and is always not the predominant factor in achieving optimal performance in downstream tasks.
	Euclidean Space	FieldE [125]; TANGO [61]	The analytical methods (e.g., ODE) facilitate the acquisition of dynamic and continuous representations of entities and relations.		

resulting in the need for additional training resources. Consequently, caution must be exercised to manage the model expense or implement a justifiable acceleration of the algorithm when utilising advanced geometric embeddings. (2). The results of pattern inference (in Section 4.3) are discussed from an algebraic perspective. The important patterns (e.g., *symmetry*, *antisymmetry*, *inversion* and *composition*) of the knowledge graphs are closely related to the algebraic operations employed by the KGE models. Table 7 reveals that most models that can simultaneously capture all four patterns gain from *Product* operations, such as Hadamard product and Hamilton product. This suggests us to develop more product-based models to capture the relational patterns of KG concurrently. Moreover, exploring innovative algebraic operations under certain conditions is also worthwhile. (3). Section 4.4 enumerates a variety of applications in which KGs can be employed across different domains, encompassing intrinsic applications such as KGQA and knowledge reasoning, as well as extrinsic applications such as leveraging KGs in Biomedicine. Concurrently, the recent rise of LLMs [17, 31, 149, 166], which are closely associated with KGs, presents an opportunity for mutual reinforcement. Nevertheless, LLMs trained on general corpora may not effectively generalise to specific domains or novel knowledge owing to the absence of domain-specific knowledge or new training data. In order to solve the above problems, a potential solution is to integrate Knowledge Graphs into LLMs because of their ability to provide accurate and explicit knowledge. It is also

well known that KGs have great symbolic reasoning capabilities and can actively evolve with the continuous input of new knowledge. Therefore, infusing KGs into other domains, enhancing LLM reasoning with knowledge graphs and interpretability, enriching knowledge graphs with LLMs, and integrating LLMs with KGs are all promising avenues for future research.

5 Future Directions

In this section, we summarise some core advantages of mathematical structures and propose several noteworthy future directions, which we hope to inspire the readers to construct more flexible and adaptable KGE algorithms in the future.

5.1 Algebraic Operation

The ability of knowledge representation is often directly related to the properties of model's algebraic operations. The algebraic advantages can sometimes compensate for the model's shortcomings, which allow the model to handle complex and non-intuitive relationships, such as 1-N, N-1 and N-M relations. One future direction is to construct a powerful operation that is able to deal with multiple tasks simultaneously. Recent works such as CompoundE [50] have started using combination operations to better handle complex relation types in different conditions. Another promising method is to build a unified algebraic KGE framework [64, 110] for better understanding the nature of algebraic operations. Nevertheless, algebraically inspired KGE models still have many perspectives to ponder.

5.2 Geometric Embedding

Several current KGE methods are inspired by some useful geometric properties. For instance, ManifoldE [198] utilise spherical embedding to extend the entity representation from one point to a sphere. ATTH [24] provides a hyperbolic embedding for capturing hierarchies based on hyperbolic isometries. However, most knowledge graphs embrace complicated structures which results in these existing methods perform poorly when modelling the knowledge graph with hybrid structures [21]. Hence, it is crucial to establish more comprehensive geometric embedding methods. Some works provide mixture-curvature embedding [186] or geometry interactive [21] KGE to enhance the ability of geometric embedding. Another meaningful research direction is how to build geometric knowledge learning models with low energy, since complex geometric embedding always needs more training time, especially on large-scale KGs. Wang et al. [179] provides a fewer parameters "RotH-like [24]" model. However, it is still a worthwhile direction to build a comprehensive geometric embedding model while keeping low energy consumption.

5.3 Analytical Optimisation

Analytical properties of knowledge representation are of great importance but poorly studied. Any of these properties, such as convergence, stability, complexity and derivability, etc., will greatly affect the model performance on downstream tasks. For example, the embedding space has to be differentiable, or the model could not be trained by gradient descent. Several works focus on solving certain analytical properties in KGE. TorusE [38] started by analysing convergence, by choosing a compact Lie group as an embedding space, the model never diverges unlimited and regularisation is no longer required. In addition, it is worth paying special attention to build some highly efficient KGE models through optimisation analytically. Peng et al. [135] adopt segmented embeddings to divide the entity representation space into multiple independent sub-spaces explicitly. Nevertheless, there still has a long way to go to deal with analytical optimisations, so as to construct more reliable KGE models in the future.

5.4 Others

It is worth noting that the above three mathematical perspectives are not independent but complementary to each other. Therefore, one possible research direction is to build multi-mathematical perspective models. FieldE [125] employs the neural ODE that represents relations as trajectories connecting neighbouring nodes in the graph, which can continuously (analytical view) represent the underlying geometries including Euclidean, Poincare Ball, Hyperboloid and Spherical (geometric view). Moreover, it is of great significance to note the above summarised implications are **not limited to the KGE domain, but indeed can be extended to any domains**. Recent works strongly bear this out. Hyperbolic space can be utilised in generative models [35] geometrically, neural order differential equations (NODE) are considered as another breakthrough in deep learning with strong abilities in supervised learning, sequential diffusion [79, 81, 164], etc. In summary, it is imperative to harness the attributes of mathematical space in order to investigate further potentialities, encompassing not only Knowledge Graph Embeddings but also extending to various other domains.

6 Conclusion

Over the recent years, the rapid growth of Knowledge Graph Embedding has been witnessed. As we have seen in our analysis of the current state of the art, most of the existing works tend to utilise the properties of different spaces to build knowledge representations. Hence, we present a comprehensive review of the state-of-the-art KGE techniques from the perspective of representation spaces, and discuss the existing KGE methods according to three mathematical perspectives: (1) Algebraic perspective, (2) Geometric perspective, and (3) Analytical perspective. In particular, we first introduce the basic concept of mathematical spaces and the relationship between them. We then introduce and compare the strengths of KGE techniques from different mathematical perspectives in terms of embedding space, scoring function, optimisation, etc. At the same time, some unique characteristics of specific spaces are summarised through the experimental results. For instance, hyperbolic-based models manifest their competence in assimilating hierarchical information. We also discuss some future directions such as efficient geometric embedding, and rational optimisation methods. Furthermore, in the context of the burgeoning big model epoch, we propose several prospective avenues for integrating LLMs and KGs. We believe that it is vital to analyse the model from the mathematical point of view, to determine more precisely what causes the possible failures in each method and find the appropriate mathematics tools to solve them. This article takes the field of KGE as an example to show the powerful role of mathematical spaces and their properties, thereby inspiring more strong mathematical modelling in more fields.

References

- [1] ABOUD, R., CEYLAN, I., LUKASIEWICZ, T., AND SALVATORI, T. Boxe: A box embedding model for knowledge base completion. *Advances in Neural Information Processing Systems* 33 (2020), 9649–9661.
- [2] AMIN, S., VARANASI, S., DUNFIELD, K. A., AND NEUMANN, G. Lower: Low-rank bilinear pooling for link prediction. In *International Conference on Machine Learning* (2020), PMLR, pp. 257–268.
- [3] AUER, S., BIZER, C., KOBILAROV, G., LEHMANN, J., CYGANIAK, R., AND IVES, Z. Dbpedia: A nucleus for a web of open data. In *The semantic web*, vol. 4825. Springer, 2007, pp. 722–735.
- [4] BALAZEVIC, I., ALLEN, C., AND HOSPEDALES, T. Multi-relational poincaré graph embeddings. *Advances in Neural Information Processing Systems* 32 (2019), 4463–4473.
- [5] BALAZEVIĆ, I., ALLEN, C., AND HOSPEDALES, T. Tucker: Tensor factorization for knowledge graph completion. In *Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing* (2019), pp. 5185–5194.
- [6] BALL, W. W. R. *A short account of the history of mathematics*. Courier Corporation, 1960.
- [7] BANG, Y., CAHYAWIJAYA, S., LEE, N., DAI, W., SU, D., WILIE, B., LOVENIA, H., JI, Z., YU, T., CHUNG, W., ET AL. A multitask, multilingual, multimodal evaluation of ChatGPT on reasoning, hallucination, and interactivity. *arXiv*

preprint [arXiv:2302.04023](https://arxiv.org/abs/2302.04023) (2023).

- [8] BASTOS, A., SINGH, K., NADGERI, A., SHEKARPOUR, S., MULANG, I. O., AND HOFFART, J. Hopfe: Knowledge graph representation learning using inverse hopf fibrations. In *International Conference on Information and Knowledge Management* (2021), pp. 89–99.
- [9] BELLINI, V., ANELLI, V. W., DI NOIA, T., AND DI SCIASCIO, E. Auto-encoding user ratings via knowledge graphs in recommendation scenarios. In *Proceedings of the 2nd Workshop on Deep Learning for Recommender Systems* (2017), pp. 60–66.
- [10] BODENREIDER, O. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research* 32, suppl_1 (2004), D267–D270.
- [11] BOLLACKER, K., EVANS, C., PARITOSH, P., STURGE, T., AND TAYLOR, J. Freebase: a collaboratively created graph database for structuring human knowledge. In *ACM SIGMOD International Conference on Management of Data* (2008), pp. 1247–1250.
- [12] BORDES, A., GLOROT, X., WESTON, J., AND BENGIO, Y. A semantic matching energy function for learning with multi-relational data. *Machine Learning* 94, 2 (2014), 233–259.
- [13] BORDES, A., USUNIER, N., GARCIA-DURAN, A., WESTON, J., AND YAKHNEKO, O. Irreflexive and hierarchical relations as translations. *arXiv preprint arXiv:1304.7158* (2013).
- [14] BORDES, A., USUNIER, N., GARCIA-DURAN, A., WESTON, J., AND YAKHNEKO, O. Translating embeddings for modeling multi-relational data. *Advances in Neural Information Processing Systems* 26 (2013).
- [15] BORDES, A., WESTON, J., COLLOBERT, R., AND BENGIO, Y. Learning structured embeddings of knowledge bases. In *AAAI Conference on Artificial Intelligence* (2011), pp. 301–306.
- [16] BOURBAKI, N. *Commutative algebra: chapters 1-7*. Springer, 1989.
- [17] BROWN, T., MANN, B., RYDER, N., SUBBIAH, M., KAPLAN, J. D., DHARIWAL, P., NEELAKANTAN, A., SHYAM, P., SASTRY, G., ASKELL, A., ET AL. Language models are few-shot learners. *Advances in Neural Information Processing Systems* 33 (2020), 1877–1901.
- [18] BURNAEV, E., AND BERNSTEIN, A. Manifold modeling in machine learning. *Journal of Communications Technology and Electronics* 66, 6 (2021), 754–763.
- [19] CAO, X., AND LIU, Y. Relmkg: reasoning with pre-trained language models and knowledge graphs for complex question answering. *Applied Intelligence* 53, 10 (2023), 12032–12046.
- [20] CAO, Z., XU, Q., YANG, Z., CAO, X., AND HUANG, Q. Dual quaternion knowledge graph embeddings. In *AAAI Conference on Artificial Intelligence* (2021), pp. 6894–6902.
- [21] CAO, Z., XU, Q., YANG, Z., CAO, X., AND HUANG, Q. Geometry interaction knowledge graph embeddings. In *AAAI Conference on Artificial Intelligence* (2022), pp. 5521–5529.
- [22] CARVALHO, R. M., OLIVEIRA, D., AND PESQUITA, C. Knowledge graph embeddings for icu readmission prediction. *BMC Medical Informatics and Decision Making* 23, 1 (2023), 12.
- [23] CHAI, J., AND SHI, G. Module: Module embedding for knowledge graphs. *arXiv preprint arXiv:2203.04702* (2022).
- [24] CHAMI, I., WOLF, A., JUAN, D.-C., SALA, F., RAVI, S., AND RÉ, C. Low-dimensional hyperbolic knowledge graph embeddings. *arXiv preprint arXiv:2005.00545* (2020).
- [25] CHAO, L., HE, J., WANG, T., AND CHU, W. Pairre: Knowledge graph embeddings via paired relation vectors. In *Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing* (2021), pp. 4360–4369.
- [26] CHEN, R. T., RUBANOVA, Y., BETTENCOURT, J., AND DUVENAUD, D. K. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems* (2018), pp. 6572–6583.
- [27] CHEN, Y., LIU, J., ZHANG, Z., WEN, S., AND XIONG, W. Möbiuse: Knowledge graph embedding on möbius ring. *Knowledge-Based Systems* 227 (2021), 107181.
- [28] CHENG, S., ZHANG, N., TIAN, B., DAI, Z., XIONG, F., GUO, W., AND CHEN, H. Editing language model-based knowledge graph embeddings. *arXiv preprint arXiv:2301.10405* (2023).
- [29] CHO, K., VAN MERRIËNBOER, B., GULCEHRE, C., BAHDANAU, D., BOUGARES, F., SCHWENK, H., AND BENGIO, Y. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).
- [30] CHOI, B., AND KO, Y. Knowledge graph extension with a pre-trained language model via unified learning method. *Knowledge-Based Systems* 262 (2023), 110245.
- [31] CUI, Y., YANG, Z., AND YAO, X. Efficient and effective text encoding for chinese llama and alpaca. *arXiv preprint arXiv:2304.08177* (2023).
- [32] DEMAILLY, J.-P. *Analytic methods in algebraic geometry*, vol. 1. International Press Somerville, MA, 2012.
- [33] DETTMERS, T., MINERVINI, P., STENETORP, P., AND RIEDEL, S. Convolutional 2d knowledge graph embeddings. In *AAAI Conference on Artificial Intelligence* (2018), pp. 1811–1818.
- [34] DEVLIN, J., CHANG, M.-W., LEE, K., AND TOUTANOVA, K. Bert: Pre-training of deep bidirectional transformers for

language understanding. *arXiv preprint arXiv:1810.04805* (2018).

- [35] DING, J., AND REGEV, A. Deep generative model embedding of single-cell rna-seq profiles on hyperspheres and hyperbolic spaces. *Nature communications* 12, 1 (2021), 1–17.
- [36] DONG, L., WEI, F., ZHOU, M., AND XU, K. Question answering over freebase with multi-column convolutional neural networks. In *Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing* (2015), pp. 260–269.
- [37] DONG, Y., GUO, X., XIANG, J., LIU, K., AND TANG, Z. Hypersphere: An embedding method for knowledge graph completion based on hypersphere. In *International Conference on Knowledge Science, Engineering and Management* (2021), pp. 517–528.
- [38] EBISU, T., AND ICHISE, R. Toruse: Knowledge graph embedding on a lie group. In *AAAI Conference on Artificial Intelligence* (2018), pp. 1819–1826.
- [39] EBISU, T., AND ICHISE, R. Generalized translation-based embedding of knowledge graph. *IEEE Transactions on Knowledge and Data Engineering* 32, 5 (2019), 941–951.
- [40] ENSAN, F., AND BAGHERI, E. Document retrieval model through semantic linking. In *ACM International Conference on Web Search and Data Mining* (2017), pp. 181–190.
- [41] FAITH, C. *Algebra II Ring Theory: Vol. 2: Ring Theory*, vol. 191. Springer Science & Business Media, 2012.
- [42] FEI, H., REN, Y., ZHANG, Y., JI, D., AND LIANG, X. Enriching contextualized language model from knowledge graph for biomedical information extraction. *Briefings in bioinformatics* 22, 3 (2021), 110.
- [43] FENG, J., HUANG, M., YANG, Y., AND ZHU, X. Gake: Graph aware knowledge embedding. In *International Conference on Computational Linguistics* (2016), pp. 641–651.
- [44] FITZPATRICK, R. *Euclid's elements of geometry*. Euclidis Elementa, 2007.
- [45] GANEA, O., BÉCIGNEUL, G., AND HOFMANN, T. Hyperbolic entailment cones for learning hierarchical embeddings. In *International Conference on Machine Learning* (2018), PMLR, pp. 1646–1655.
- [46] GAO, C., SUN, C., SHAN, L., LIN, L., AND WANG, M. Rotate3d: Representing relations as rotations in three-dimensional space for knowledge graph embedding. In *International Conference on Information and Knowledge Management* (2020), pp. 385–394.
- [47] GAO, H., YANG, K., YANG, Y., ZAKARI, R. Y., OWUSU, J. W., AND QIN, K. Quatte: Dynamic quaternion embedding for knowledge graph completion. *arXiv preprint arXiv:2105.09002* (2021).
- [48] GAO, L., ZHU, H., ZHUO, H. H., AND XU, J. Dual quaternion embeddings for link prediction. *Applied Sciences* 11, 12 (2021), 5572.
- [49] GAUR, M., GUNARATNA, K., SRINIVASAN, V., AND JIN, H. Iseeq: Information seeking question generation using dynamic meta-information retrieval and knowledge graphs. In *AAAI Conference on Artificial Intelligence* (2022), pp. 10672–10680.
- [50] GE, X., WANG, Y.-C., WANG, B., AND KUO, C.-C. J. Compound: Knowledge graph embedding with translation, rotation and scaling compound operations. *arXiv preprint arXiv:2207.05324* (2022).
- [51] GE, X., WANG, Y.-C., WANG, B., AND KUO, C.-C. J. Knowledge graph embedding with 3d compound geometric transformations. *arXiv preprint arXiv:2304.00378* (2023).
- [52] GE, X., WANG, Y.-C., WANG, B., AND KUO, C. J. Core: A knowledge graph entity type prediction method via complex space regression and embedding. *Pattern Recognition Letters* 157 (2022), 97–103.
- [53] GEORGI, H., AND GLASHOW, S. L. Unity of all elementary-particle forces. *Physical Review Letters* 32, 8 (1974), 438.
- [54] GOLDMAN, W. M. varieties of representations. In *Geometry of Group Representations: Proceedings of the AMS-IMS-SIAM Joint Summer Research Conference* (1988), p. 169.
- [55] GREGUCCI, C., NAYYERI, M., HERNÁNDEZ, D., AND STAAB, S. Link prediction with attention applied on multiple knowledge graph embedding models. In *International Conference on World Wide Web* (2023), pp. 2600–2610.
- [56] GREUB, W. H. *Linear algebra*, vol. 23. Springer Science & Business Media, 2012.
- [57] GU, A., SALA, F., GUNEL, B., AND RÉ, C. Learning mixed-curvature representations in product spaces. In *International Conference for Learning Representation* (2018).
- [58] GUAN, N., SONG, D., AND LIAO, L. Knowledge graph embedding with concepts. *Knowledge-Based Systems* 164 (2019), 38–44.
- [59] GUO, J., AND KOK, S. Bique: Biquaternionic embeddings of knowledge graphs. In *Conference on Empirical Methods in Natural Language Processing* (2021), pp. 8338–8351.
- [60] GUO, L., SUN, Z., AND HU, W. Learning to exploit long-term relational dependencies in knowledge graphs. In *International Conference on Machine Learning* (2019), PMLR, pp. 2505–2514.
- [61] HAN, Z., DING, Z., MA, Y., GU, Y., AND TRESP, V. Temporal knowledge graph forecasting with neural ode. *arXiv preprint arXiv:2101.05151* (2021).
- [62] HAO, J., CHEN, M., YU, W., SUN, Y., AND WANG, W. Universal representation learning of knowledge bases by jointly embedding instances and ontological concepts. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*

(2019), pp. 1709–1719.

- [63] HARRIS, J., AND STOCKER, H. Spherical geometry. *Handbook of mathematics and computational science* (1998), 108–113.
- [64] HAYASHI, K., AND SHIMBO, M. On the equivalence of holographic and complex embeddings for link prediction. *arXiv preprint arXiv:1702.05563* (2017).
- [65] HE, S., LIU, K., JI, G., AND ZHAO, J. Learning to represent knowledge graphs with gaussian embedding. In *International Conference on Information and Knowledge Management* (2015), pp. 623–632.
- [66] HOCHREITER, S., AND SCHMIDHUBER, J. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.
- [67] HOFFMANN, R., ZHANG, C., LING, X., ZETTLEMOYER, L., AND WELD, D. S. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies* (2011), pp. 541–550.
- [68] HOGAN, A., BLOMQUIST, E., COCHEZ, M., D’AMATO, C., MELO, G. D., GUTIERREZ, C., KIRrane, S., GAYO, J. E. L., NAVIGLI, R., NEUMAIER, S., NGOMO, A.-C. N., POLLERES, A., RASHID, S. M., RULA, A., SCHMELZEISEN, L., SEQUEDA, J., STAAB, S., AND ZIMMERMANN, A. Knowledge graphs. *ACM Comput. Surv.* 54, 4 (2021).
- [69] HØST, A. M., LISON, P., AND MOONEN, L. Constructing a knowledge graph from textual descriptions of software vulnerabilities in the national vulnerability database. *arXiv preprint arXiv:2305.00382* (2023).
- [70] HOUSEHOLDER, A. S. Unitary triangularization of a nonsymmetric matrix. *Journal of the ACM* 5, 4 (1958), 339–342.
- [71] HUANG, X., TANG, J., TAN, Z., ZENG, W., WANG, J., AND ZHAO, X. Knowledge graph embedding by relational and entity rotation. *Knowledge-Based Systems* 229 (2021), 107310.
- [72] HUANG, Z., WANG, D., HUANG, B., ZHANG, C., SHANG, J., LIANG, Y., WANG, Z., LI, X., FALOUTSOS, C., SUN, Y., ET AL. Concept2box: Joint geometric embeddings for learning two-view knowledge graphs. *arXiv preprint arXiv:2307.01933* (2023).
- [73] IYER, R. G., BAI, Y., WANG, W., AND SUN, Y. Dual-geometric space embedding model for two-view knowledge graphs. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (2022), pp. 676–686.
- [74] IZACARD, G., AND GRAVE, E. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EAACL* (2021), pp. 874–880.
- [75] JI, G., HE, S., XU, L., LIU, K., AND ZHAO, J. Knowledge graph embedding via dynamic mapping matrix. In *Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing* (2015), pp. 687–696.
- [76] JI, G., LIU, K., HE, S., AND ZHAO, J. Knowledge graph completion with adaptive sparse transfer matrix. In *AAAI Conference on Artificial Intelligence* (2016), pp. 985–991.
- [77] JI, S., PAN, S., CAMBRIA, E., MARTTINEN, P., AND PHILIP, S. Y. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems* 33, 2 (2022), 494–514.
- [78] JI, Z., LEE, N., FRIESKE, R., YU, T., SU, D., XU, Y., ISHII, E., BANG, Y. J., MADOTTO, A., AND FUNG, P. Survey of hallucination in natural language generation. *ACM Computing Surveys* 55, 12 (2023), 1–38.
- [79] JIANG, B., ZHANG, Y., WEI, X., XUE, X., AND FU, Y. Learning compositional representation for 4d captures with neural ode. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), pp. 5340–5350.
- [80] JIANG, Z., XU, F. F., ARAKI, J., AND NEUBIG, G. How can we know what language models know. *Trans. Assoc. Comput. Linguistics* 8 (2020), 423–438.
- [81] KANG, Q., SONG, Y., DING, Q., AND TAY, W. P. Stable neural ode with lyapunov-stable equilibrium points for defending against adversarial attacks. *Advances in Neural Information Processing Systems* (2021), 14925–14937.
- [82] KAZEMI, S. M., AND POOLE, D. Simple embedding for link prediction in knowledge graphs. *Advances in Neural Information Processing Systems* 31 (2018).
- [83] KIPF, T. N., AND WELLING, M. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).
- [84] KOBYZEV, I., PRINCE, S. J., AND BRUBAKER, M. A. Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43, 11 (2020), 3964–3979.
- [85] LACROIX, T., USUNIER, N., AND OBOZINSKI, G. Canonical tensor decomposition for knowledge base completion. In *International Conference on Machine Learning* (2018), pp. 2863–2872.
- [86] LAN, Y., AND JIANG, J. Query graph generation for answering multi-hop complex questions from knowledge bases. In *Annual Meeting of the Association for Computational Linguistics* (2020), pp. 969–974.
- [87] LANG, S. *Algebra*, vol. 211. Springer Science & Business Media, 2012.
- [88] LE, T., HUYNH, N., AND LE, B. Knowledge graph embedding by projection and rotation on hyperplanes for link prediction. *Applied Intelligence* 53, 9 (2023), 10340–10364.
- [89] LE, T., TRAN, H., AND LE, B. Knowledge graph embedding with the special orthogonal group in quaternion space for link prediction. *Knowledge-Based Systems* 266 (2023), 110400.
- [90] LEE, J. *Introduction to topological manifolds*, vol. 202. Springer Science & Business Media, 2010.

- [91] LEWIS, M., LIU, Y., GOYAL, N., GHAZVININEJAD, M., MOHAMED, A., LEVY, O., STOYANOV, V., AND ZETTMLOYER, L. BART : denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL (2020)*, pp. 7871–7880.
- [92] LI, G., SUN, Z., HU, W., CHENG, G., AND QU, Y. Position-aware relational transformer for knowledge graph embedding. *IEEE Transactions on Neural Networks and Learning Systems (2023)*.
- [93] LI, J., TANG, T., ZHAO, W. X., NIE, J.-Y., AND WEN, J.-R. Pretrained language models for text generation: A survey. *arXiv preprint arXiv:2201.05273 (2022)*.
- [94] LI, L., WANG, P., YAN, J., WANG, Y., LI, S., JIANG, J., SUN, Z., TANG, B., CHANG, T.-H., WANG, S., ET AL. Real-world data medical knowledge graph: construction and applications. *Artificial Intelligence in Medicine 103 (2020)*, 101817.
- [95] LI, R., ZHAO, J., LI, C., HE, D., WANG, Y., LIU, Y., SUN, H., WANG, S., DENG, W., SHEN, Y., ET AL. House: Knowledge graph embedding with householder parameterization. In *International Conference on Machine Learning (2022)*, PMLR, pp. 13209–13224.
- [96] LI, Y., KONG, D., ZHANG, Y., TAN, Y., AND CHEN, L. Robust deep alignment network with remote sensing knowledge graph for zero-shot and generalized zero-shot remote sensing image scene classification. *ISPRS Journal of Photogrammetry and Remote Sensing 179 (2021)*, 145–158.
- [97] LI, Z., JI, J., FU, Z., GE, Y., XU, S., CHEN, C., AND ZHANG, Y. Efficient non-sampling knowledge graph embedding. In *International Conference on World Wide Web (2021)*, pp. 1727–1736.
- [98] LI, Z., LIU, X., WANG, X., LIU, P., AND SHEN, Y. Transo: a knowledge-driven representation learning method with ontology information constraints. *International Conference on World Wide Web 26, 1 (2023)*, 297–319.
- [99] LI, Z.-X., LI, Y.-J., LIU, Y.-W., LIU, C., AND ZHOU, N.-X. K-ctia: Automatic analysis of cyber threat intelligence based on a knowledge graph. *Symmetry 15, 2 (2023)*, 337.
- [100] LIANG, C., BERANT, J., LE, Q., FORBUS, K. D., AND LAO, N. Neural symbolic machines: Learning semantic parsers on freebase with weak supervision. *arXiv preprint arXiv:1611.00020 (2016)*.
- [101] LIANG, S. Collaborative, dynamic and diversified user profiling. In *Proceedings of the AAAI Conference on Artificial Intelligence (2019)*, vol. 33, pp. 4269–4276.
- [102] LIANG, S. Unsupervised semantic generative adversarial networks for expert retrieval. In *The world wide web conference (2019)*, pp. 1039–1050.
- [103] LIANG, S., LUO, Y., AND MENG, Z. Profiling users for question answering communities via flow-based constrained co-embedding model. *ACM Transactions on Information Systems 40, 2 (2021)*, 1–38.
- [104] LIANG, S., TANG, S., MENG, Z., AND ZHANG, Q. Cross-temporal snapshot alignment for dynamic networks. *IEEE Transactions on Knowledge and Data Engineering (2021)*.
- [105] LIANG, S., YILMAZ, E., AND KANOULAS, E. Dynamic clustering of streaming short documents. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (2016)*, pp. 995–1004.
- [106] LIAO, S., LIANG, S., MENG, Z., AND ZHANG, Q. Learning dynamic embeddings for temporal knowledge graphs. In *ACM International Conference on Web Search and Data Mining (2021)*, pp. 535–543.
- [107] LIN, B. Y., CHEN, X., CHEN, J., AND REN, X. Kagnet: Knowledge-aware graph networks for commonsense reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP (2019)*, pp. 2829–2839.
- [108] LIN, Y., LIU, Z., LUAN, H., SUN, M., RAO, S., AND LIU, S. Modeling relation paths for representation learning of knowledge bases. *arXiv preprint arXiv:1506.00379 (2015)*.
- [109] LIN, Y., LIU, Z., SUN, M., LIU, Y., AND ZHU, X. Learning entity and relation embeddings for knowledge graph completion. In *AAAI Conference on Artificial Intelligence (2015)*, pp. 2181–2187.
- [110] LIU, H., WU, Y., AND YANG, Y. Analogical inference for multi-relational embeddings. In *International Conference on Machine Learning (2017)*, pp. 2168–2178.
- [111] LIU, X., AND FANG, H. Latent entity space: a novel retrieval approach for entity-bearing queries. *Information Retrieval Journal 18, 6 (2015)*, 473–503.
- [112] LONG, J., CHEN, Z., HE, W., WU, T., AND REN, J. An integrated framework of deep learning and knowledge graph for prediction of stock price trend: An application in chinese stock exchange market. *Applied Soft Computing 91 (2020)*, 106205.
- [113] LOU, A., KATSMAN, I., JIANG, Q., BELONGIE, S., LIM, S.-N., AND DE SA, C. Differentiating through the fréchet mean. In *International Conference on Machine Learning (2020)*, pp. 6393–6403.
- [114] LU, F., CONG, P., AND HUANG, X. Utilizing textual information in knowledge graph embedding: A survey of methods and applications. *IEEE Access 8 (2020)*, 92072–92088.
- [115] LU, H., AND HU, H. Dense: An enhanced non-abelian group representation for knowledge graph embedding. *arXiv preprint arXiv:2008.04548 (2020)*.
- [116] LV, X., HOU, L., LI, J., AND LIU, Z. Differentiating concepts and instances for knowledge graph embedding. *arXiv preprint arXiv:1811.04588 (2018)*.

- [117] MA, T., LI, M., LV, S., ZHU, F., HUANG, L., AND HU, S. Conte: contextualized knowledge graph embedding for circular relations. *Data Mining and Knowledge Discovery* 37, 1 (2023), 110–135.
- [118] MARCHEGGIANI, D. AND TITOV, I. Encoding sentences with graph convolutional networks for semantic role labeling. In *Conference on Empirical Methods in Natural Language Processing* (2017), pp. 1506–1515.
- [119] MENG, Z., LIU, F., CLARK, T. H., SHAREGHI, E., AND COLLIER, N. Mixture-of-partitions: Infusing large biomedical knowledge graphs into bert. *arXiv preprint arXiv:2109.04810* (2021).
- [120] MENG, Z., LIU, F., CLARK, T. H., SHAREGHI, E., AND COLLIER, N. Mixture-of-partitions: Infusing large biomedical knowledge graphs into BERT. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP* (2021), pp. 4672–4681.
- [121] MENG, Z., LIU, F., SHAREGHI, E., SU, Y., COLLINS, C., AND COLLIER, N. Rewire-then-probe: A contrastive recipe for probing biomedical knowledge of pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL* (2022), pp. 4798–4810.
- [122] MOHAMED, S. K., NOUNU, A., AND NOVÁČEK, V. Biological applications of knowledge graph embedding models. *Briefings in Bioinformatics* 22, 2 (2021), 1679–1693.
- [123] MURPHY, K. P. *Machine learning: a probabilistic perspective*. MIT press, 2012.
- [124] NAYYERI, M., XIONG, B., MOHAMMADI, M., AKTER, M. M., ALAM, M. M., LEHMANN, J., AND STAAB, S. Knowledge graph embeddings using neural ito process: From multiple walks to stochastic trajectories. In *Findings of the Association for Computational Linguistics* (2023), pp. 7165–7179.
- [125] NAYYERI, M., XU, C., HOFFMANN, F., ALAM, M. M., LEHMANN, J., AND VAHDATI, S. Knowledge graph representation learning using ordinary differential equations. In *Conference on Empirical Methods in Natural Language Processing* (2021), pp. 9529–9548.
- [126] NICKEL, M., AND KIELA, D. Poincaré embeddings for learning hierarchical representations. *Advances in Neural Information Processing Systems* 30 (2017), 6338–6347.
- [127] NICKEL, M., AND KIELA, D. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *International Conference on Machine Learning* (2018), PMLR, pp. 3779–3788.
- [128] NICKEL, M., MURPHY, K., TRESP, V., AND GABRILOVICH, E. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE* 104, 1 (2015), 11–33.
- [129] NICKEL, M., ROSASCO, L., AND POGGIO, T. Holographic embeddings of knowledge graphs. In *AAAI Conference on Artificial Intelligence* (2016), pp. 1955–1961.
- [130] NICKEL, M., TRESP, V., AND KRIEDEL, H.-P. A three-way model for collective learning on multi-relational data. In *International Conference on Machine Learning* (2011), pp. 809–816.
- [131] PAN, S., LUO, L., WANG, Y., CHEN, C., WANG, J., AND WU, X. Unifying large language models and knowledge graphs: A roadmap. *arXiv preprint arXiv:2306.08302* (2023).
- [132] PAN, Z., AND WANG, P. Hyperbolic hierarchy-aware knowledge graph embedding for link prediction. In *Findings of Conference on Empirical Methods in Natural Language Processing* (2021), pp. 2941–2948.
- [133] PAVLOVIĆ, A., AND SALLINGER, E. Expressive: A spatio-functional embedding for knowledge graph completion. *arXiv preprint arXiv:2206.04192* (2022).
- [134] PENG, W., VARANKA, T., MOSTAFA, A., SHI, H., AND ZHAO, G. Hyperbolic deep neural networks: A survey. *arXiv preprint arXiv:2101.04562* (2021).
- [135] PENG, X., CHEN, G., LIN, C., AND STEVENSON, M. Highly efficient knowledge graph embedding learning with orthogonal procrustes analysis. *arXiv preprint arXiv:2104.04676* (2021).
- [136] PENG, Y., AND ZHANG, J. Linear: Simple but powerful knowledge graph embedding for link prediction. In *IEEE International Conference on Data Mining* (2020), IEEE, pp. 422–431.
- [137] PETRONI, F., ROCKTÄSCHEL, T., RIEDEL, S., LEWIS, P. S. H., BAKHTIN, A., WU, Y., AND MILLER, A. H. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP* (2019), pp. 2463–2473.
- [138] PFLAUM, M. *Analytic and geometric study of stratified spaces: contributions to analytic and geometric aspects*. No. 1768. Springer Science & Business Media, 2001.
- [139] QIU, Y., WANG, Y., JIN, X., AND ZHANG, K. Stepwise reasoning for multi-relation question answering over knowledge graph with weak supervision. In *ACM International Conference on Web Search and Data Mining* (2020), pp. 474–482.
- [140] RAFFEL, C., SHAZEER, N., ROBERTS, A., LEE, K., NARANG, S., MATENA, M., ZHOU, Y., LI, W., AND LIU, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research* 21, 1 (2020), 5485–5551.
- [141] RAVIV, H., KURLAND, O., AND CARMEL, D. Document retrieval using entity-based language models. In *ACM SIGIR Conference on Research and Development in Information Retrieval* (2016), pp. 65–74.
- [142] ROBINSON, G. D. B. On the representations of the symmetric group. *American Journal of Mathematics* (1938), 745–760.
- [143] ROSSET, C., XIONG, C., PHAN, M., SONG, X., BENNETT, P., AND TIWARY, S. Knowledge-aware language model pretraining.

- arXiv preprint arXiv:2007.00655* (2020).
- [144] ROTMAN, J. J. *A first course in abstract algebra*. Pearson College Division, 2000.
- [145] RUDIN, W. *Functional analysis*, 1973.
- [146] SAFAVI, T., KOUTRA, D., AND MEIJ, E. Evaluating the calibration of knowledge graph embeddings for trustworthy link prediction. In *Empirical Methods in Natural Language Processing* (2020), pp. 8308–8321.
- [147] SALA, F., DE SA, C., GU, A., AND RÉ, C. Representation tradeoffs for hyperbolic embeddings. In *International Conference on Machine Learning* (2018), pp. 4460–4469.
- [148] SAXENA, A., TRIPATHI, A., AND TALUKDAR, P. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *Annual Meeting of the Association for Computational Linguistics* (2020), pp. 4498–4507.
- [149] SCAO, T. L., FAN, A., AKIKI, C., PAVLICK, E., ILIĆ, S., HESSLOW, D., CASTAGNÉ, R., LUCCIONI, A. S., YVON, F., GALLÉ, M., ET AL. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100* (2022).
- [150] SCHLICHTKRULL, M., KIPF, T. N., BLOEM, P., VAN DEN BERG, R., TITOV, I., AND WELLING, M. Modeling relational data with graph convolutional networks. In *European Semantic Web Conference* (2018), Springer, pp. 593–607.
- [151] SHANG, C., TANG, Y., HUANG, J., BI, J., HE, X., AND ZHOU, B. End-to-end structure-aware convolutional networks for knowledge base completion. In *AAAI Conference on Artificial Intelligence* (2019), vol. 33, pp. 3060–3067.
- [152] SHEN, T., MAO, Y., HE, P., LONG, G., TRISCHLER, A., AND CHEN, W. Exploiting structured knowledge in text via graph-guided representation learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP* (2020).
- [153] SHEN, W., WANG, J., LUO, P., AND WANG, M. Linden: linking named entities with knowledge base via semantic knowledge. In *International Conference on World Wide Web* (2012), pp. 449–458.
- [154] SOCHER, R., CHEN, D., MANNING, C. D., AND NG, A. Reasoning with neural tensor networks for knowledge base completion. *Advances in Neural Information Processing Systems* (2013), 926–934.
- [155] SUCHANEK, F. M., KASNECI, G., AND WEIKUM, G. Yago: a core of semantic knowledge. In *International Conference on World Wide Web* (2007), pp. 697–706.
- [156] SUN, R., CAO, X., ZHAO, Y., WAN, J., ZHOU, K., ZHANG, F., WANG, Z., AND ZHENG, K. Multi-modal knowledge graphs for recommender systems. In *International Conference on Information and Knowledge Management* (2020), pp. 1405–1414.
- [157] SUN, Z., CHEN, M., HU, W., WANG, C., DAI, J., AND ZHANG, W. Knowledge association with hyperbolic knowledge graph embeddings. In *Conference on Empirical Methods in Natural Language Processing* (2020), pp. 5704–5716.
- [158] SUN, Z., DENG, Z.-H., NIE, J.-Y., AND TANG, J. Rotate: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint arXiv:1902.10197* (2019).
- [159] SUNG, M., LEE, J., YI, S., JEON, M., KIM, S., AND KANG, J. Can language models be biomedical knowledge bases? *arXiv preprint arXiv:2109.07154* (2021).
- [160] SUZUKI, A., ENOKIDA, Y., AND YAMANISHI, K. Riemannian transe: Multi-relational graph embedding in non-euclidean space.
- [161] TABACOF, P., AND COSTABELLO, L. Probability calibration for knowledge graph embedding models. In *International Conference on Learning Representations* (2019).
- [162] TANG, Y., HUANG, J., WANG, G., HE, X., AND ZHOU, B. Orthogonal relation transforms with graph context modeling for knowledge graph embedding. In *Annual Meeting of the Association for Computational Linguistics* (2020), pp. 2713–2722.
- [163] TAY, Y., TUAN, L. A., AND HUI, S. C. Hyperbolic representation learning for fast and efficient neural question answering. In *ACM International Conference on Web Search and Data Mining* (2018), pp. 583–591.
- [164] THORPE, M., NGUYEN, T. M., XIA, H., STROHMER, T., BERTOZZI, A., OSHER, S., AND WANG, B. Grand++: Graph neural diffusion with a source term. In *International Conference for Learning Representation* (2021).
- [165] TOUTANOVA, K., AND CHEN, D. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd workshop on continuous vector space models and their compositionality* (2015), pp. 57–66.
- [166] TOUVRON, H., LAVRIL, T., IZACARD, G., MARTINET, X., LACHAUX, M.-A., LACROIX, T., ROZIÈRE, B., GOYAL, N., HAMBRO, E., AZHAR, F., ET AL. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* (2023).
- [167] TREVES, F. *Topological Vector Spaces, Distributions and Kernels: Pure and Applied Mathematics, Vol. 25*, vol. 25. Elsevier, 2016.
- [168] TROUILLON, T., WELBL, J., RIEDEL, S., GAUSSIER, É., AND BOUCHARD, G. Complex embeddings for simple link prediction. In *International Conference on Machine Learning* (2016), pp. 2071–2080.
- [169] UNGAR, A. A. Hyperbolic trigonometry and its application in the poincaré ball model of hyperbolic geometry. *Computers & Mathematics with Applications* 41, 1-2 (2001), 135–147.
- [170] VASHISHTH, S., SANYAL, S., NITIN, V., AND TALUKDAR, P. Composition-based multi-relational graph convolutional networks. *arXiv preprint arXiv:1911.03082* (2019).
- [171] VASWANI, A., SHAZEER, N., PARMAR, N., USZKOREIT, J., JONES, L., GOMEZ, A. N., KAISER, Ł., AND POLOSUKHIN, I. Attention is all you need. *Advances in Neural Information Processing Systems* 30 (2017).
- [172] VILELA, J., ASIF, M., MARQUES, A. R., SANTOS, J. X., RASGA, C., VICENTE, A., AND MARTINIANO, H. Biomedical

- knowledge graph embeddings for personalized medicine: Predicting disease-gene associations. *Expert Systems* 40, 5 (2023), e13181.
- [173] VILNIS, L., AND MCCALLUM, A. Word representations via gaussian embedding. *arXiv preprint arXiv:1412.6623* (2014).
- [174] VRANDEČIĆ, D., AND KRÖTZSCH, M. Wikidata: a free collaborative knowledgebase. *Communications of the ACM* 57, 10 (2014), 78–85.
- [175] WAN, G., AND DU, B. Gaussianpath: A bayesian multi-hop reasoning framework for knowledge graph reasoning. In *AAAI Conference on Artificial Intelligence* (2021), vol. 35, pp. 4393–4401.
- [176] WANG, B., MENG, Q., WANG, Z., WU, D., CHE, W., WANG, S., CHEN, Z., AND LIU, C. Interht: Knowledge graph embeddings by interaction between head and tail entities. *arXiv preprint arXiv:2202.04897* (2022).
- [177] WANG, F., ZHANG, Z., SUN, L., YE, J., AND YAN, Y. Dirie: knowledge graph embedding with dirichlet distribution. In *International Conference on World Wide Web* (2022), pp. 3082–3091.
- [178] WANG, H., ZHANG, F., ZHAO, M., LI, W., XIE, X., AND GUO, M. Multi-task feature learning for knowledge graph enhanced recommendation. In *International Conference on World Wide Web* (2019), pp. 2000–2010.
- [179] WANG, K., LIU, Y., LIN, D., AND SHENG, M. Hyperbolic geometry is not necessary: Lightweight euclidean-based models for low-dimensional knowledge graph embeddings. In *Findings of Conference on Empirical Methods in Natural Language Processing* (2021), pp. 464–474.
- [180] WANG, P., LI, S., AND PAN, R. Incorporating gan for negative sampling in knowledge representation learning. In *AAAI Conference on Artificial Intelligence* (2018), vol. 32.
- [181] WANG, Q., MAO, Z., WANG, B., AND GUO, L. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering* 29, 12 (2017), 2724–2743.
- [182] WANG, Q., WANG, B., AND GUO, L. Knowledge base completion using embeddings and rules. In *International Joint Conference on Artificial Intelligence* (2015).
- [183] WANG, S., FU, K., SUN, X., ZHANG, Z., LI, S., AND JIN, L. Hierarchical-aware relation rotational knowledge graph embedding for link prediction. *Neurocomputing* 458 (2021), 259–270.
- [184] WANG, S., LIU, R., SHEN, L., AND KHATTAK, A. M. Stke: Temporal knowledge graph embedding in the spherical coordinate system. In *International Conference on Artificial Intelligence and Security* (2022), pp. 292–305.
- [185] WANG, S., WEI, X., DOS SANTOS, C. N., WANG, Z., NALLAPATI, R., ARNOLD, A., AND PHILIP, S. Y. Knowledge graph representation via hierarchical hyperbolic neural graph embedding. In *IEEE International Conference on Big Data* (2021), pp. 540–549.
- [186] WANG, S., WEI, X., NOGUEIRA DOS SANTOS, C. N., WANG, Z., NALLAPATI, R., ARNOLD, A., XIANG, B., YU, P. S., AND CRUZ, I. F. Mixed-curvature multi-relational graph neural network for knowledge graph completion. In *International Conference on World Wide Web* (2021), pp. 1761–1771.
- [187] WANG, X., GAO, T., ZHU, Z., ZHANG, Z., LIU, Z., LI, J., AND TANG, J. KEPLER: A unified model for knowledge embedding and pre-trained language representation. *Trans. Assoc. Comput. Linguistics* 9 (2021), 176–194.
- [188] WANG, X., GAO, T., ZHU, Z., ZHANG, Z., LIU, Z., LI, J., AND TANG, J. Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics* 9 (2021), 176–194.
- [189] WANG, X., HE, X., CAO, Y., LIU, M., AND CHUA, T.-S. Kgat: Knowledge graph attention network for recommendation. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (2019), pp. 950–958.
- [190] WANG, Z., ZHANG, J., FENG, J., AND CHEN, Z. Knowledge graph embedding by translating on hyperplanes. In *AAAI Conference on Artificial Intelligence* (2014), pp. 1112–1119.
- [191] WEBER, M., AND NICKEL, M. Curvature and representation learning: Identifying embedding spaces for relational data. *Advances in Neural Information Processing Systems* (2018).
- [192] WILKINSON, J. H., BAUER, F. L., AND REINSCH, C. *Linear algebra*, vol. 2. Springer, 2013.
- [193] WISE, C., CALVO, M. R., BHATIA, P., IOANNIDIS, V., KARYPUS, G., PRICE, G., SONG, X., BRAND, R., AND KULKANI, N. Covid-19 knowledge graph: Accelerating information retrieval and discovery for scientific literature. In *Proceedings of Knowledgeable NLP: the First Workshop on Integrating Structured Knowledge and Neural Networks for NLP* (2020), pp. 1–10.
- [194] WU, F., AND WELD, D. S. Open information extraction using wikipedia. In *Annual Meeting of the Association for Computational Linguistics* (2010), pp. 118–127.
- [195] WU, Y., LU, B., TIAN, L., AND LIANG, S. Learning to co-embed queries and documents. *Electronics* 11, 22 (2022), 3694.
- [196] XIAO, H., HUANG, M., HAO, Y., AND ZHU, X. Transa: An adaptive approach for knowledge graph embedding. *arXiv preprint arXiv:1509.05490* (2015).
- [197] XIAO, H., HUANG, M., HAO, Y., AND ZHU, X. Transg: A generative mixture model for knowledge graph embedding. *arXiv preprint arXiv:1509.05488* (2015).
- [198] XIAO, H., HUANG, M., AND ZHU, X. From one point to a manifold: Knowledge graph embedding for precise link prediction. *arXiv preprint arXiv:1512.04792* (2015).

- [199] XIAO, H., LIU, X., SONG, Y., WONG, G. Y., AND SEE, S. Complex hyperbolic knowledge graph embeddings with fast fourier transform. *arXiv preprint arXiv:2211.03635* (2022).
- [200] XIE, R., LIU, Z., JIA, J., LUAN, H., AND SUN, M. Representation learning of knowledge graphs with entity descriptions. In *AAAI Conference on Artificial Intelligence* (2016), vol. 30.
- [201] XIE, R., LIU, Z., LUAN, H., AND SUN, M. Image-embodied knowledge representation learning. In *International Joint Conference on Artificial Intelligence* (2017), pp. 3140–3146.
- [202] XIE, R., LIU, Z., SUN, M., ET AL. Representation learning of knowledge graphs with hierarchical types. In *Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing* (2016), pp. 2965–2971.
- [203] XIE, R., LIU, Z., SUN, M., ET AL. Representation learning of knowledge graphs with hierarchical types. In *International Joint Conference on Artificial Intelligence* (2016), vol. 2016, pp. 2965–2971.
- [204] XIONG, B., ZHU, S., NAYYERI, M., XU, C., PAN, S., ZHOU, C., AND STAAB, S. Ultrahyperbolic knowledge graph embeddings. *arXiv preprint arXiv:2206.00449* (2022).
- [205] XIONG, W., HOANG, T., AND WANG, W. Y. Deeppath: A reinforcement learning method for knowledge graph reasoning. *arXiv preprint arXiv:1707.06690* (2017).
- [206] XU, C., AND LI, R. Relation embedding with dihedral group in knowledge graph. In *Annual Meeting of the Association for Computational Linguistics* (2019), pp. 263–272.
- [207] XU, K., REDDY, S., FENG, Y., HUANG, S., AND ZHAO, D. Question answering on freebase via relation extraction and textual evidence. *arXiv preprint arXiv:1603.00957* (2016).
- [208] XU, M. Understanding graph embedding methods and their applications. *SIAM Review* 63, 4 (2021), 825–853.
- [209] YANG, B., YIH, W.-T., HE, X., GAO, J., AND DENG, L. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575* (2014).
- [210] YANG, H., AND LIU, J. Knowledge graph representation learning as groupoid: unifying transe, rotate, quate, complex. In *International Conference on Information and Knowledge Management* (2021), pp. 2311–2320.
- [211] YANG, T., SHA, L., AND HONG, P. Nage: Non-abelian group embedding for knowledge graphs. In *International Conference on Information and Knowledge Management* (2020), pp. 1735–1742.
- [212] YAO, L., MAO, C., AND LUO, Y. Kg-bert: Bert for knowledge graph completion. *arXiv preprint arXiv:1909.03193* (2019).
- [213] YAO, T., WANG, Y., ZHANG, K., AND LIANG, S. Improving the expressiveness of k-hop message-passing gnnns by injecting contextualized substructure information. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (2023), pp. 3070–3081.
- [214] YASUNAGA, M., REN, H., BOSSELUT, A., LIANG, P., AND LESKOVEC, J. Qa-gnn: Reasoning with language models and knowledge graphs for question answering. *arXiv preprint arXiv:2104.06378* (2021).
- [215] YASUNAGA, M., REN, H., BOSSELUT, A., LIANG, P., AND LESKOVEC, J. QA-GNN: reasoning with language models and knowledge graphs for question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT* (2021), pp. 535–546.
- [216] YIH, S. W.-T., CHANG, M.-W., HE, X., AND GAO, J. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *Joint Conference of the Annual Meeting of the ACL and International Joint Conference on Natural Language Processing of the AFNLP* (2015), pp. 1321–1331.
- [217] YU, D., ZHU, C., YANG, Y., AND ZENG, M. Jaket: Joint pre-training of knowledge graph and language understanding. In *AAAI Conference on Artificial Intelligence* (2022), pp. 11630–11638.
- [218] YU, J., CAI, Y., SUN, M., AND LI, P. Mquade: a unified model for knowledge fact embedding. In *International Conference on World Wide Web* (2021), pp. 3442–3452.
- [219] YU, L., LUO, Z., LIU, H., LIN, D., LI, H., AND DENG, Y. Triplere: Knowledge graph embeddings via tripled relation vectors. *arXiv preprint arXiv:2209.08271* (2022).
- [220] ZHANG, C., YAO, H., HUANG, C., JIANG, M., LI, Z., AND CHAWLA, N. V. Few-shot knowledge graph completion. In *AAAI Conference on Artificial Intelligence* (2020), pp. 3041–3048.
- [221] ZHANG, Q., WANG, R., YANG, J., AND XUE, L. Knowledge graph embedding by translating in time domain space for link prediction. *Knowledge-Based Systems* 212 (2021), 106564.
- [222] ZHANG, Q., WANG, R., YANG, J., AND XUE, L. Knowledge graph embedding by reflection transformation. *Knowledge-Based Systems* 238 (2022), 107861.
- [223] ZHANG, Q., WANG, R., YANG, J., AND XUE, L. Structural context-based knowledge graph embedding for link prediction. *Neurocomputing* 470 (2022), 109–120.
- [224] ZHANG, R., HRISTOVSKI, D., SCHUTTE, D., KASTRIN, A., FISZMAN, M., AND KILICOGULU, H. Drug repurposing for covid-19 via knowledge graph completion. *Journal of biomedical informatics* 115 (2021), 103696.
- [225] ZHANG, S., TAY, Y., YAO, L., AND LIU, Q. Quaternion knowledge graph embeddings. *Advances in Neural Information Processing Systems* (2019), 2731–2741.
- [226] ZHANG, W., CHEN, T., WANG, J., AND YU, Y. Optimizing top-n collaborative filtering via dynamic negative item

- sampling. In *ACM SIGIR Conference on Research and Development in Information Retrieval* (2013), pp. 785–788.
- [227] ZHANG, W., PAUDEL, B., ZHANG, W., BERNSTEIN, A., AND CHEN, H. Interaction embeddings for prediction and explanation in knowledge graphs. In *ACM International Conference on Web Search and Data Mining* (2019), pp. 96–104.
- [228] ZHANG, X., BOSSELUET, A., YASUNAGA, M., REN, H., LIANG, P., MANNING, C. D., AND LESKOVEC, J. Greaselm: Graph reasoning enhanced language models. In *International Conference on Learning Representations* (2022).
- [229] ZHANG, X., YANG, Q., AND XU, D. Trans: Transition-based knowledge graph embedding with synthetic relation representation. *arXiv preprint arXiv:2204.08401* (2022).
- [230] ZHANG, Y., YAO, Q., AND CHEN, L. Interstellar: Searching recurrent architecture for knowledge graph embedding. *Advances in Neural Information Processing Systems* 33 (2020), 10030–10040.
- [231] ZHANG, Z., CAI, J., ZHANG, Y., AND WANG, J. Learning hierarchy-aware knowledge graph embeddings for link prediction. In *AAAI Conference on Artificial Intelligence* (2020), pp. 3065–3072.
- [232] ZHANG, Z., HAN, X., LIU, Z., JIANG, X., SUN, M., AND LIU, Q. Ernie: Enhanced language representation with informative entities. In *Annual Meeting of the Association for Computational Linguistics* (2019), pp. 1441–1451.
- [233] ZHANG, Z., LIU, X., ZHANG, Y., SU, Q., SUN, X., AND HE, B. Pretrain-kge: learning knowledge representation from pretrained language models. In *Findings of the Association for Computational Linguistics* (2020), pp. 259–266.
- [234] ZHANG, Z., PARULIAN, N., JI, H., ELSAYED, A., MYERS, S., AND PALMER, M. Fine-grained information extraction from biomedical literature based on knowledge-enriched abstract meaning representation. In *Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing* (2021), pp. 6261–6270.
- [235] ZHANG, Z., ZHUANG, F., QU, M., LIN, F., AND HE, Q. Knowledge graph embedding with hierarchical relation structure. In *Conference on Empirical Methods in Natural Language Processing* (2018), pp. 3198–3207.
- [236] ZHAO, H., YAO, Q., LI, J., SONG, Y., AND LEE, D. L. Meta-graph based recommendation fusion over heterogeneous information networks. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (2017), pp. 635–644.
- [237] ZHENG, W., WANG, W., QIAN, F., ZHAO, S., AND ZHANG, Y. Hyperbolic hierarchical knowledge graph embeddings for link prediction in low dimensions. *arXiv preprint arXiv:2204.13704* (2022).
- [238] ZHENG, W., YIN, L., CHEN, X., MA, Z., LIU, S., AND YANG, B. Knowledge base graph embedding module design for visual question answering model. *Pattern Recognition* 120 (2021), 108153.
- [239] ZHOU, K., ZHAO, W. X., BIAN, S., ZHOU, Y., WEN, J.-R., AND YU, J. Improving conversational recommender systems via knowledge graph based semantic fusion. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (2020), pp. 1006–1014.