Two-dimensional Taxonomy for N-ary Knowledge Representation Learning Methods

Xiaohua Lu, Liubov Tupikina, and Mehwish Alam

***Abstract*—Real-world knowledge can take various forms, including structured, semi-structured, and unstructured data. Among these, knowledge graphs are a form of structured human knowledge that integrate heterogeneous data sources into structured representations but typically reduce complex n-ary relations to simple triples, thereby losing higher-order relational details. In contrast, hypergraphs naturally represent n-ary relations with hyperedges, which directly connect multiple entities together. Yet hypergraph representation learning often overlooks entity roles in hyperedges, limiting the fine-grained semantic modelling. To address these issues, knowledge hypergraphs and hyper-relational knowledge graphs combine the advantages of knowledge graphs and hypergraphs to better capture the complex structures and role-specific semantics of real-world knowledge. This survey provides a comprehensive review of methods handling n-ary relational data, covering both knowledge hypergraphs and hyper-relational knowledge graphs literatures. We propose a two-dimensional taxonomy: the first dimension categorises models based on their methodology, i.e., translation-based models, tensor factorisation-based models, deep neural network-based models, logic rules-based models, and hyperedge expansion-based models. The second dimension classifies models according to their awareness of entity roles and positions in n-ary relations, dividing them into aware-less, position-aware, and role-aware approaches. Finally, we discuss existing datasets, training settings and strategies, and outline open challenges to inspire future research.**

***Index Terms*—N-ary relational representation learning, n-ary relatinoal data, multi-arity relation, hyper-relational fact, knowledge hypergraph, knowledge hypergraph representation learning, hyper-relational knowledge graph, hyper-relational knowledge graph representation learning, survey, taxonomy.**

I. INTRODUCTION

WITH the increasing interest in knowledge graphs (KGs), which is synonymous with knowledge base (KBs) [1], in areas such as education [2], scientific research [3], social sciences [4], healthcare [5], and finance [6], the design of knowledge bases capable of supporting complex reasoning and semantic modelling has become a key research topic [7]. KGs represent knowledge in a structured triple, such as $\langle$head entity, relation, tail entity$\rangle$, the semantic descriptions of entities can also be considered as literals [8]. Figure 1(a) show an example of KG related to the film *Inception*. KGs support a variety of large-scale open-source KBs, including WordNet [9], DBpedia [10], YAGO [11], Freebase [12], and Wikidata [13]. These KBs are often built using Semantic Web standards, such as the Resource Description Framework (RDF) and the Web Ontology Language (OWL). These standards define the structure of data and relationships in KGs, facilitating the integration of heterogeneous data sources and supporting the construction of machine-interpretable knowledge structures [7].

Xiaohua Lu (Corresponding author) is with the University of Rouen Normandy, Rouen, France (e-mail: xh.lu0706@gmail.com).

Liubov Tupikina is with Nokia Bell Labs, Paris, France.

Mehwish Alam is with Télécom Paris, Institut Polytechnique de Paris, Paris, France.

Manuscript received June 27, 2025; revised xx xx, 2025.

To enable downstream tasks such as KG completion [14], [15], question answering [16]–[18], recommender systems [19], and reasoning [20], [21], KG representation learning (KGRL), also called KG embedding, has emerged as a key technique. It aims to encode entities and relations into low-dimensional vectors while preserving their semantic and structural properties [7]. However, due to the inheritance structure of KGs, KGRL models can only handle binary relations, which are insufficient to capture the multi-entity interactions commonly observed in the real world.

In practice, many facts involve more than two entities in a relation, forming n-ary interactions, also called the higher-order relations [22]. To overcome this limitation, hypergraphs (HGs) have been introduced as a natural extension of traditional graphs. Unlike standard graphs where edges connect only two nodes, HGs allow each hyperedge to connect multiple nodes simultaneously, allowing direct representation of n-ary interactions. For example, a co-authored paper often involves multiple authors [23]–[26], as shown in Figure 1(b), and a group in a social network may include many members [27], [28]. Accordingly, hypergraph representation learning (HGRL) has become an important approach to derive meaningful embeddings for nodes or hyperedges. Despite recent advances, achieving scalable and efficient structural integration remains a key challenge due to high computational cost and complex model design [29]–[32].

To bridge this gap, knowledge hypergraphs (KHGs) [33] and hyper-relational knowledge graphs (HKGs) [34] have been proposed, combining the expressive structure of HGs with the semantic modeling capabilities of KGs. KHGs use directed, labeled hyperedges to encode n-ary facts, as illustrated in Figure 2(b), while HKGs enrich the primary triples with qualifier pairs, as shown in Figure 2(a). These representations mitigate the drawbacks of star-to-clique transformation or reification techniques, that convert n-ary facts into multiple binary forms. These transformations often suffer from limitations, including loss of higher-order structures, increased knowledge sparsity, and over-parameterisation problems [33]. Therefore, the direct modeling of n-ary facts via KHGs and HKGs has received increasing attention in recent years [19], [33], [34].

Additionally, real-world knowledge is inherently dynamic. Temporal Knowledge Graphs (TKGs) have been proposed

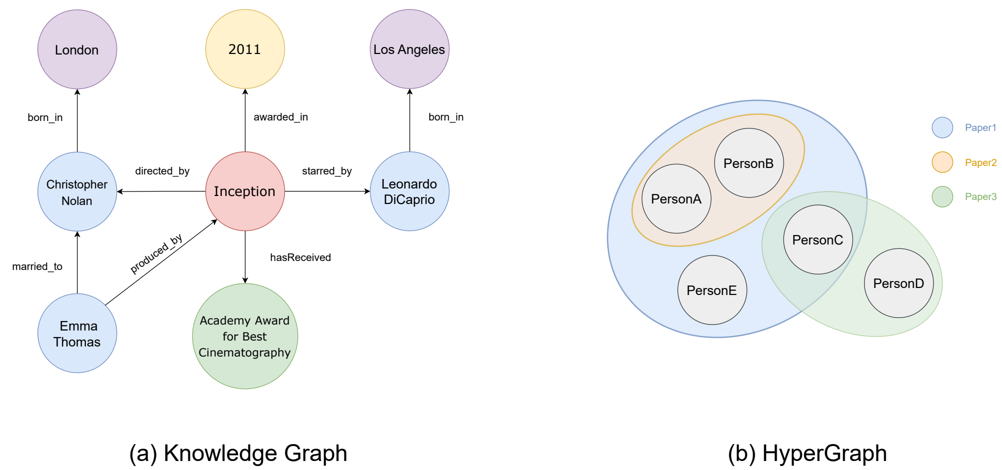


Fig. 1. Comparison of KG and HG. (a) A standard KG models facts about the film Inception using binary relations. (b) A HG represents co-authorship in a research network, capturing higher-order group interactions via hyperedges.

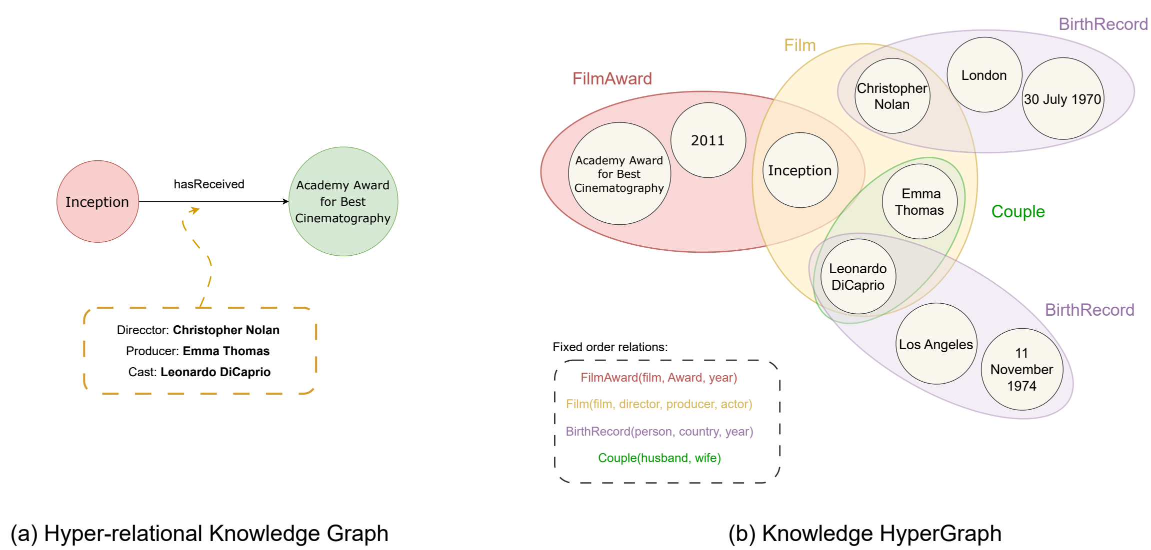


Fig. 2. Comparison of KHG and HKG. (a) A HKG enriches primary triples with a set of role-specific qualifiers. (b) A KHG directly models n-ary relational facts such as Film, FilmAward, BirthRecord and Couple using fixed-ordered tuples. These relations consist of entities with semantic roles assigned to specific positions.

to capture time-evolving facts and address this issue. TKGs augment each triple-based fact with a timestamp, forming a quadruple (h, r, t, τ), where τ denotes the time of occurrence [35]. There exist some knowledge bases, such as Wikidata [13] and YAGO2 [36], integrate time annotations into factual data. Despite growing interest in TKGs, temporal KHGs remain largely unexplored, which provides a promising direction for future work.

In particular, some knowledge hypergraph representation learning (KHGRL) models are direct extensions of existing KGRL models, e.g., m-TransH [37], m-CP [33], or of HGRL models, e.g., G-MPNN [38], H^{2}GNN [39]. Despite this, the field remains relatively fragmented. To address this, we propose a unified two-dimensional taxonomy that systematically categorises n-ary relational representation learning methods, with the aim of facilitating deeper understanding and guiding future research in this emerging area. Specifically, our taxonomy organises existing models based on (i) their technical methodologies, i.e., tensor factorisation-based, translation-based, deep neural network-based, logic rule-based, hyperedge expansion-based, and (ii) their levels of semantic awareness, i.e., position-aware, role-aware, and aware-less. This classification allows for a more fine-grained understanding of the modeling capabilities of current models.

*a) **Contributions.**:* Our contributions can be summarised as follows:

- We unify the terminology and formalism for n-ary relational data, including KHGs, HKGs, hyper-relational TKGs (HTKGs), and n-tuple TKGs (N-TKGs), and clarify the corresponding link prediction tasks.
- We propose a novel two-dimensional taxonomy that highlights the evolution of n-ary relational representation learning models from both technical and semantic awareness perspectives.

- We identify the key characteristics of an effective n-ary relational representation learning model and formulate a set of design principles for future model development.
- We summarise existing benchmark datasets and provide a comparative analysis of negative sampling strategies from KGs to KHGs, offering practical guidance for evaluation and training.
- Finally, we outline future directions for n-ary relational data modeling, including dynamic reasoning, multimodal extension, and end-to-end architectures for scalable and interpretable learning.

*b) **Article organisation.**:* Section II introduces the notations and definitions for modeling n-ary knowledge data and models. Section III reviews existing surveys in the fields of KGRL and HGRL, highlights their classification strategies, and identifies the lack of a unified perspective on n-ary knowledge representation. Section IV provides a quick review of prior work on KGRL and HGRL methods. Section V presents our two-dimensional taxonomy and detailed classification of existing models. Section VI summarises some available datasets and discusses negative sampling strategies. Finally, Section VII concludes the paper and outlines future research opportunities.

II. Preliminaries

This section introduces the fundamental concepts and notations used throughout this survey. We formally define the KGs [1], [8], HGs [40], KHGs [41], and HKGs [42]. We also introduce N-TKGs [43] and HTKGs [44] as two temporal extensions of KHGs and HKGs, respectively. These structures enrich static knowledge representations by incorporating temporal attributes. We further define the link prediction tasks, also called KG completion task [45], tailored to both KHG [41] and HKG settings [46]. Finally, to describe the expressiveness of link prediction models over n-ary relational data, we introduce the notion of full expressiveness [33]. A fully expressive model can capture and represent different relation patterns through embeddings of n-ary facts. In contrast, models lacking full expressiveness can only capture limited knowledge with strong prior assumptions [47], potentially leading to unwarranted inferences [48].

Definition 1 (**Hypergraph**). A HG is a generalisation of a conventional graph defined as $\mathcal{H} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a finite set of vertices (or nodes) and $\mathcal{E}$ is a finite set of hyperedges. Each hyperedge is a non-empty subset of $\mathcal{V}$, containing arbitrary vertices. In addition, the HG is associated with an incidence matrix $\mathcal{I} \in \{0,1\}^{|\mathcal{V}|\times|\mathcal{E}|}$ defined by

$$\mathcal{I}(v, e) = [\![v \in e]\!], \tag{1}$$

where $v \in \mathcal{V}$, $e \in \mathcal{E}$ and $[\![p]\!]$ is the indicator function which equals to 1 if the predicate p is true and 0 otherwise.

Definition 2 (**Knowledge Graph**). A KG is defined as a directed labeled graph $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{L}, \mathcal{F})$, where $\mathcal{E}$ is a finite set of entities representing real or abstract concepts; $\mathcal{R}$ is a finite set of relations between entities; $\mathcal{L}$ is a finite set of literals, such as numerical values, texts, or images; and $\mathcal{F} \subseteq \mathcal{E} \times \mathcal{R} \times (\mathcal{E} \cup \mathcal{L})$ is a finite set of facts. Each fact $f \in \mathcal{F}$ is expressed as a triple (h, r, t) or as (h, r, l), where $h, t \in \mathcal{E}$, $r \in \mathcal{R}$, and $l \in \mathcal{L}$.

Definition 3 (**Knowledge Hypergraph**). A KHG is defined as $\mathcal{K} = (\mathcal{E}, \mathcal{R}, \mathcal{T}_{\mathcal{O}})$, which generalises the concept of a KG by allowing the relations to have arbitrary entities. Here, $\mathcal{E}$ is a finite set of entities, $\mathcal{R}$ is a finite set of relations (each possibly with a different arity), and $\mathcal{T}_{\mathcal{O}}$ is a finite set of observed tuples. Each tuple $t \in \mathcal{T}_{\mathcal{O}}$ is expressed as

$$t = r(\rho_r^1 : e_1, \rho_r^2 : e_2, \ldots, \rho_r^\alpha : e_\alpha), \tag{2}$$

where $r \in \mathcal{R}$ is a relation of arity α (with $\alpha \geq 1$), each $e_i \in \mathcal{E}$ is an entity, and ρ_r^i denotes the role of e_i in the relation r. Furthermore, each tuple t can be associated with a truth indicator b_t, where $b_t = 1$ indicates that t is true and $b_t = 0$ otherwise. If $\alpha = 2$ for all relations, the KHG reduces to the standard KG.

Definition 4 (**Hyper-relational Knowledge Graph**). A HKG is defined as $\mathcal{HKG} = (\mathcal{V}, \mathcal{R}, \mathcal{E})$, where $\mathcal{V}$ is a finite set of entities, $\mathcal{R}$ is a finite set of relations, and $\mathcal{E}$ is a list of hyper-relational facts. Each hyper-relational fact is represented as a tuple $(s, r, o, \mathcal{Q})$, where $(s, r, o) \in \mathcal{V} \times \mathcal{R} \times \mathcal{V}$ denotes the primary triple. $\mathcal{Q} \subseteq \mathcal{P}(\mathcal{R} \times \mathcal{V})$ is a set of qualifier pairs $(a_i : v_i)$, with $a_i \in \mathcal{R}$ and $v_i \in \mathcal{V}$. The set $\mathcal{P}(\cdot)$ denotes the power set. We use $\mathcal{Q}_i^j$ to refer to the i_{th} qualifier pair associated with fact f_j.

Definition 5 (**Hyper-relational Temporal Knowledge Graph**). Let $\mathcal{E}$, $\mathcal{R}$, and $\mathcal{T}$ denote the finite sets of entities, relations, and timestamps, respectively. A HTKG is defined as a set of time-stamped hyper-relational facts. Each fact is represented as:

$$t = ((s, r, o, t), (a_i, v_i)_{i=1}^n), \tag{3}$$

where (s, r, o, t) is the primary quadruple, and each qualifier (a_i, v_i) consists of a qualifier relation $a_i \in \mathcal{R}$ and a qualifier entity $v_i \in \mathcal{E}$. The integer n denotes the number of qualifiers.

Definition 6 (**N-tuple Temporal Knowledge Graphs**). An N-TKG is a sequence of timestamped KGs $\mathcal{G} = \{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_t\}$. Each $\mathcal{G}_t = (\mathcal{V}_{\text{pred}}, \mathcal{V}_{\text{ent}}, \mathcal{V}_\rho, \mathcal{F}_t)$, where $\mathcal{V}_{\text{pred}}$ is a finite set of predicates, $\mathcal{V}_{\text{ent}}$ is a finite set of entities, $\mathcal{V}_\rho$ is a finite set of roles, and $\mathcal{F}_t$ is a set of facts at time t. A fact $f \in \mathcal{F}_t$ is expressed as:

$$f = (\text{pred}, \rho_1{:}e_1, \ldots, \rho_n{:}e_n, t), \tag{4}$$

where pred is a predicate, e_i is an entity, ρ_i is its role, and t is the timestamp.

Definition 7 (**Link Prediction for KHG**). In a KHG, an n-ary fact is represented as an n-ary tuple:

$$t = r(\rho_r^1 : e_1, \rho_r^2 : e_2, \ldots, \rho_r^\alpha : e_\alpha), \tag{5}$$

where r is a relation, e_i are entities, and ρ_r^i denotes the role of e_i in relation r. Let $\mathcal{T}_O \subseteq \mathcal{T}_T \subseteq \mathcal{T}$ denote the observed tuples, the ground-truth tuples, and all possible n-ary tuples, respectively. The set of hidden tuples is defined as $\mathcal{T}_H = \mathcal{T} \setminus \mathcal{T}_O$. The goal of link prediction in KHGs is to predict the labels of hidden tuples $\mathcal{T}_H$, given the observed tuples $\mathcal{T}_O$, i.e., to identify which unobserved tuples are likely to be true. For example, given the incomplete n-ary fact: $r(e_1, e_2, ?, e_4)$, the goal is to predict the missing object entity e_3.

Definition 8 (**Link Prediction for HKG**). A hyper-relational fact typically consists of a primary triple (s, r, o) together with a set of qualifier pairs $\{(a_i : v_i)\}_{i=1}^{m}$, where s, o, v_i are entities and r, a_i are relations. n-ary link prediction in hyper-relational facts aims to infer missing elements from an incomplete fact. The missing part can be either an entity (e.g., $s, o, v_1, \ldots, v_m$) or a relation (e.g., $r, a_1, \ldots, a_m$). For example, given the incomplete hyper-relational fact: $(s, r, ?), \{(a_i : v_i)\}_{i=1}^{m}$, the goal is to predict the missing object entity o.

Definition 9 (**Full Expressiveness**). A link prediction model is *fully expressive* if, for any ground truth over all entities and relations, there exist entity and relation embeddings that accurately distinguish true n-ary relational facts from false ones.

III. Related Works

In recent years, a number of surveys have been published to systematise the advances in KGRL and HGRL methods. In the field of KGRL, existing surveys typically classify models based on technical methodologies or embedding paradigms. For example, Wang et al. [49] distinguish between translation-based models, semantic matching-based models, and auxiliary information-based models. Ji et al. [1] and Peng et al. [7] further categorise KGRL methods by learning paradigms, covering embedding spaces, scoring functions, encoding models, and auxiliary information-enhanced models. A more recent work by Cao et al. [50] introduce a taxonomy based on mathematical perspective, dividing models into algebraic structure (e.g., ring and vector space), geometric structure (e.g., euclidean, hyperbolic and spherical geometry), and analytic structure (e.g., euclidean and probability space). The surveys [1], [7], [49] adopt similar classification criteria. Although the survey by Cao et al. [50] defines its categories from a different perspective, it is mainly based on different embedding spaces. While these surveys provide a comprehensive overview of standard KGRL techniques, they are all restricted to binary relational facts and do not address n-ary relational modelling.

In parallel, the field of HGRL has received increasing attention. Gao et al. [51] classify models by task types, including transductive HG learning, inductive HG learning, HG structure updating, and multimodal HG learning. Zhang et al. [52] introduce a theory-based taxonomy that includes matrix decomposition methods, random walk methods, and deep learning approaches. Antelmi et al. [40] organise models by technical perspective, namely spectral-based methods, proximity-preserving methods, and deep neural network-based methods. Tian et al. [22] focus on higher-order network representations, including network motifs, simplicial complexes, and HGs. They categorise HGRL techniques into tensor factorisation-based models, hypergraph cut-based models, random walk-based models, and hypergraph reconstruction-based models. The surveys [22], [40], [52] share similar technical categories. Although survey [51] is task-oriented, the underlying modelling techniques align closely with the others. While these surveys successfully outline the modelling landscape for HGs, they do not consider the complex semantic challenges of multi-relational knowledge modelling [40].

To the best of our knowledge, no existing survey has yet targeted representation learning over KHGs or HKGs, both of which aim to model complex n-ary relational facts that cannot be effectively captured by binary triples or simple hyperedges. This paper addresses this gap by proposing a novel **two-dimensional taxonomy** for n-ary knowledge representation learning. Our taxonomy classifies models based on two orthogonal dimensions: (i) their technical modelling principles, and (ii) their level of semantic awareness (e.g., position-aware, role-aware, aware-less). By combining perspectives from both KGRL and HGRL, this survey provides a unified view of n-ary relational modelling, and establishes a foundation for future research on KHG and HKG embeddings.

IV. Technical Background

KGRL and HGRL have enabled substantial progress in modelling multi-relational data. Many KHGRL and HKGRL methods extend the foundational techniques from KGRL or HGRL to accommodate n-ary relational structures. For example, m-TransH [37], m-CP [33], and HSimplE [33] are extensions of KGRL methods, while G-MPNN [38] and H^{2}GNN [39] originate from HGRL designs.

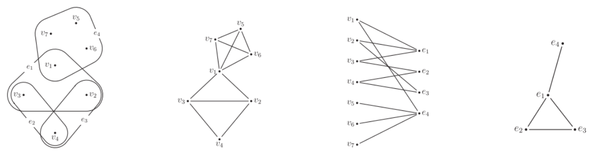
To better understand the technical foundations of KHGRL and HKGRL models, this section briefly reviews the key modelling strategies developed in the KGRL and HGRL literatures. On the one hand, these approaches form the basis of the first dimension of our proposed two-dimensional taxonomy: the modelling techniques. On the other hand, these quick reviews help to identify how existing approaches have been transferred and where existing ideas have not yet been fully explored. For example, the random walk strategies are widely used in HGRL, such as Hyper-gram [53] and Lbsn2vec++ [54]. However, they have not yet been fully exploited in KHGRL, which is a potential area for future research.

In summary, this section provides a concise technical overview of KGRL and HGRL model families to support the methodological foundation for the two-dimensional taxonomy proposed in Section V.

A. HGRL Methods

This section provides a technical overview of existing HGRL models, including **HG reconstruction**, **matrix (tensor) decomposition-based models**, **random walk-based models**, and **deep learning-based models**. It serves as a part of technical foundation for the technical developments discussed in Section V.

1) HG Reconstruction: HG reconstruction methods allow the transformation of higher-order structures into conventional graphs, so that existing graph-based frameworks can be efficiently reused and reduce the computational complexity [40]. A common approach is clique expansion [55], also known as two-section graphs [40], see Figure 3(b), where each hyperedge is transformed into pairwise subsets to form a clique. Other transformation approaches include star expansion and line graphs [56]. Star expansion, also called the incidence graph, represents a HG as a bipartite graph, where one set of nodes corresponds to entities and the other to hyperedges, as shown in Figure 3(c). An edge connects a node and a hyperedge if the node belongs to that hyperedge. A line graph transforms each hyperedge into a node, and connects two nodes if the corresponding hyperedges have at least one common entity, see Figure 3(d). Although these transformations simplify the higher-order relations, they may introduce structural redundancy during reconstruction and lose inherent structural information from the original HGs [40].



(a) Hypergraph. (b) Two-section Graph (a.k.a clique graph). (c) Incidence graph (a.k.a. star expansion). (d) Line graph.

Fig. 3. Reconstructions of HG to graph. [40]

2) Matrix Factorisation-based Methods: Matrix factorisation-based HGRL, also called Tensor factorisation, aims to construct an approximate matrix of the original HG and then decompose it to learn low-dimensional node embeddings. These approaches fall into two main categories: Laplacian matrix factorisation and adjacency matrix factorisation [52]. **Laplacian-based methods**, also called spectral methods, use the eigenvalues and eigenvectors of the HG Laplacian matrix to perform structural-only spectral embeddings [57]–[65]. In order to further encode the nodes with additional features, the vector-attribute-aware spectral embedding methods are proposed by extending the structural-only spectral embedding methods [66]–[70].

Adjacency matrix decomposition methods aim to learn vector embeddings by minimising objective functions, e.g., least squares, to approximate node similarity by dot product or cosine similarity [54], [71]–[75]. Despite their effectiveness, these approaches often suffer from limited scalability due to their high spatial and temporal complexity [52]. Moreover, they typically ignore second-order proximity and provide an algebraic, rather than a geometric, interpretation of the HG structure [40].

3) Random Walk-based Methods: Random walk-based HGRL methods capture both global and local structural patterns by randomly sampling paths from the HGs. Classical approaches often adopt probabilistic models such as Skip-gram or Bag-of-Words [76], and have been widely used in HGRL settings [24], [26], [53], [77]–[81]. Random sampling strategies can use *Depth-First Search (DFS)*, as in DeepWalk [82], which better captures global structures. Alternatively, *Breadth-First Search (BFS)* is used in methods such as LINE [83], which focuses on local neighbourhood structures. Node2vec [84] combines both DFS and BFS strategies to balance global and local structure capture, resulting in higher quality and more informative samples.

4) Deep Neural Network-based Methods: Deep neural network-based HGRL methods leverage deep neural networks to model higher-order dependencies and node features. These approaches fall into four major categories: message-passing frameworks, attention-based models, spectral convolutional approaches, and encoder-decoder architectures. **Message-passing HG networks**, e.g., [25], [85]–[96], extend traditional GNNs to aggregate information across hyperedges. To enhance model expressiveness, (multi-head) **attention mechanisms** have been widely applied [95], [97]–[105], allowing more flexible neighbour weighting and improved discriminative power. **Spectral convolution** methods [23], [87], [106], [107] use HG Laplacians or wavelet transforms to model complex HG topology. **Encoder-decoder** architectures, e.g., [26], [80], [108]–[110] are also explored for HGRL to learn low-dimensional embeddings, although they often underperform in structural representation tasks.

B. KGRL Methods

Similar to the motivation of Section IV-A, this section presents a technical overview of KGRL methods, covering **translation-based methods**, **encoding models** and **auxiliary information-based models**. It aims to provide some theoretical foundations that can be used in our proposed unified taxonomy.

1) Translation-based Models: Translation-based KGRL methods embed entities and relations in continuous vector spaces and model relational patterns via translation operations. These approaches aim to capture relation properties such as symmetry, asymmetry, inversion, and composition [50].

Point vector space models [111]–[120] project entities and relations into Cartesian, polar, or spherical coordinate systems. Despite their simplicity and effectiveness, these methods struggle to represent complex relation patterns. To address this, several works propose the use of **complex vector spaces** [121]–[130], which allow rotation-based operations to model symmetric and antisymmetric interactions more effectively.

Manifold-based approaches [131]–[135] are introduced to overcome the rigid geometry constraints of earlier vector-space models [1] by embedding entities in more flexible manifolds to improve semantic diversity.

Hyperbolic space, known for its ability to capture hierarchical structures, provides a natural space for representing tree-like knowledge [119], [133], [136]–[143]. However, it has limitations in logical reasoning and constant curvature assumptions.

Group-based embedding methods [144]–[149] use algebraic groups (e.g., Lie and dihedral groups) to perform geometric transformations such as rotation, reflection, and inversion, allowing more expressive modelling of relational patterns.

2) Encoding Models: Encoding models learn semantic representations by modelling interactions between entities and relations through specific architectures, including linear/bilinear mappings, tensor factorisations, and neural networks. **Linear and bilinear** models encode entity-relation interactions via linear or bilinear transformations, projecting head entities towards tail entities. This family includes representative models, such as SimplE, DistMult, RESCAL, etc. [150]–[155] **Tensor factorisation** methods treat KGRL as a third-order tensor decomposition task. Typical works include LFM, TuckER, etc. [156]–[158] **Neural networks** learn deep semantic interactions between entities and relations through non-linear transformations. MLP-based models [159]–[161] encode semantic matching patterns. CNN-based models [162]–[164] improve feature representation using convolutional filters. RNN-based frameworks [165]–[167] model long-term dependencies in relational paths. Transformer-based models [168], [169] improve context-aware representation learning, while GNN-based encoders [170]–[173] explicitly leverage graph structures for relational encoding. Generative adversarial methods such as KBGAN [174] further improve training by refining the negative sampling process.

3) Auxiliary information-enhanced Models: Auxiliary information-enhanced KGRL methods enhance entity and relation representations by incorporating external and multimodal data, such as textual descriptions, type constraints, relation paths, and visual information. Some models [175]–[178] align structured triple embeddings with unstructured textual information (e.g., entity names or descriptions) to enhance representation learning. Other works [179]–[182] incorporate semantic constraints such as entity or relation types. Visual-enhanced embeddings have also been explored by integrating image features [183]. Relation path-based approaches [184] consider multi-hop relational paths, while temporal-aware models [185] utilise temporal information of evolving KGs to enrich semantic embeddings. Additionally, uncertainty-aware models [186]–[189] assign confidence scores to facts and calibrate probability distributions, preserving both semantic and uncertainty information.

V. Two-dimensional Taxonomy for N-ary Relational Representation Learning Models

To better understand and systematise existing approaches, we propose a two-dimensional taxonomy for n-ary relational representation learning methods. The first dimension categorises models by model techniques or principles, such as tensor factorisation-based methods, translation-based methods, deep neural network-based methods, and logic rules and hyperedge expansion-based methods. These approaches differ significantly in how they encode the semantic and relational information of n-ary facts.

- **Translation-based methods.** Inspired by TransE-style models, these methods project entities and relations into a shared space and embed relations through entity projections and translations. For n-ary facts, scoring functions are extended over multiple entities.
- **Tensor factorisation-based methods.** Also known as tensor decomposition methods. These models represent n-ary relational facts as higher-order tensors and approximate them using low-rank decompositions, such as CP or Tucker. The goal is to capture multi-interactions between entities while reducing the complexity of the n-ary representation.
- **Deep neural network-based methods.** These models leverage learnable architectures, e.g., Multilayer Perceptrons (MLPs), Convolutional Neural Networks (CNNs), Graph Neural Networks (GNNs), Hypergraph Neural Networks (HNNs), and Transformers to encode n-ary facts, often using shared or fact-level interaction functions. Their flexibility allows the modelling of arbitrary arity facts and complex relational patterns.
- **Logic rules and hyperedge expansion-based methods.** These models explore symbolic reasoning or structure transformation techniques to encode entity-role semantics, improving the interpretability of the model and modelling more complex hierarchical structures.

The second dimension reflects the degree to which models are aware of entity roles and entity positions in n-ary facts. On this basis we define three categories: position-aware models, role-aware models, and aware-less models.

- **Position-aware models** incorporate the relative or absolute position of each entity, typically via position embeddings or importance/weight of entities in n-ary facts.
- **Role-aware models** go further by encoding entities and their specific roles in the relation, allowing fine-grained semantic representation of n-ary facts.
- **Aware-less models** do not explicitly encode position or role information of entities within an n-ary facts. These models treat entities as an unordered tuple, reflecting only semantic information of entities and minimal higher-order structural information.

This two-dimensional taxonomy allows for a structured comparison of existing n-ary relational models. These distinctions guide both theoretical understanding and empirical comparisons. An overview of n-ary relational representation models under this taxonomy is presented in Table I.

A. Position-aware Models

Position-aware models incorporate information about the position or importance of entities within an n-ary fact. By using position encoding, dedicated positional transformation functions or entity weights, these models can implicitly learn the role contributions through position-aware embeddings.

1) Translation-based Methods: In the n-ary setting, translation-based position-aware models explicitly distinguish positions of entities during RL. **m-TransH** [37] extends TransH to n-ary relational modelling by redefining an n-ary fact as a mapping of *roles* to entities. For each relation r, it introduces two orthogonal unit vectors n_r and b_r, and a function $a_r(\rho)$ to weight each role ρ. The scoring function is defined as Eq 6:

$$f_r(t) = \Big| \sum_{\rho \in \mathcal{M}(R_r)} a_r(\rho), P_{n_r}(t(\rho)) + b_r \Big|^2, \tag{6}$$

where $P_{n_r}(z)$ denotes the projection of the vector z onto the hyperplane n_r. However, m-TransH is not fully expressive (see Def 9), which limits its modelling capacity [42], [150], [193].

To overcome this limitation, **BoxE** [191] is proposed as a fully expressive translation-based model. It represents entities as points and relations as hyper-rectangles (or boxes). Each

TABLE I
TWO-DIMENSIONAL TAXONOMY FOR n-ARY RELATIONAL REPRESENTATION LEARNING MODELS.

	Role-aware	**Position-aware**	**Aware-less**
Translation-based		m-TransH [37] BoxE [191]	RAE [190]
Tensor Factorisation-based	RAM [192] PosKHG [41]	m-CP [33] HSimplE [33] HypE [33] ReAIE [193]	r-SimplE [33] m-DistMult [33] GETD [47] S2S [194]
Deep Neural Network-based	NaLP [195] RHKH [198] HINGE [201] NeuInfer [34] STARE [42] STARQE [205] GRAN [46] HyNT [206] QUAD [45] Hy-Transformer [207] ShrinkE [208] HyperMono [209] SDK [210] HAHE [211] HyperFormer [212] HyperTKG [44] NE-Net [43]	HyConvE [196] LPACN [199] G-MPNN [38] H^{2}GNN [39] HySAE [204]	ZHANG et al. [197] KHG-Aclair [200] HyCubE [202] HyperQuery [203]
Logic Rule-based	HyperMLN [213]		
Hyperedge Expansion-based	TransEQ [214]		

entity is parameterised by a base vector $\mathbf{e}_i$ and a translation vector $\mathbf{b}_i$, while an n-ary fact is valid if each translated entity embedding falls within its corresponding box. The model supports logical properties such as symmetry and inversion, but has limitations in capturing complex patterns [213].

2) Tensor factorisation-based Models: Tensor factorisation-based position-aware models explicitly assign position embeddings to entities, allowing the entity representations to contain semantic information at different locations in the n-ary relation. **m-CP** [33] generalises the CP decomposition [215] by assigning α different embeddings to each entity, where α is the maximum arity in the dataset. The scoring function aggregates different positional embeddings as Eq 7:

$$\phi(r(e_1,\ldots,e_\alpha)) = \odot(\mathbf{r},\mathbf{e_1},\ldots,\mathbf{e}_\alpha). \tag{7}$$

While m-CP distinguishes between entity positions, it treats the same entity differently depending on its location, and it is unable to generalise to mixed-arity datasets [46].

To mitigate the lack of shared positional information of the entity, **HSimplE** [33] assigns only one shared embedding to each entity. The model generates an embedding $\mathbf{e_i}$ for each entity, then shift $\mathbf{e_i}$ using a **shift**$(\mathbf{e},x)$ function to encode the semantic bias at different positions. This **shift** operator is relatively simple and does not introduce any additional parameters, but the hard-coded shift is not capable of learning a rich semantic information. **HypE** [33] further generalises this design by replacing the shift function with a learnable convolutional operator $\mathbf{f}(\mathbf{e},i)$. For a tuple $(r,e_1,\ldots,e_n)$, the scoring function is as Eq 8:

$$\phi(r(e_1,\ldots,e_n)) = \odot(\mathbf{r},\mathbf{f}(\mathbf{e_1},1),\ldots,\mathbf{f}(\mathbf{e_n},n)). \tag{8}$$

This method is is fully expressive and improves flexibility in modelling positional variations.

Since relational algebra provides a formal foundation for reasoning in relational databases [216], it can enhance the ability of a model to perform complex logical reasoning over n-ary facts. Although a model is fully expressive, it does not mean that it can fully support relational algebra reasoning [193]. To address this limitation, **ReAIE** [193] proposes a fully expressive model with support for relational algebra reasoning, including a large subset of primitive relational algebra operations, namely renaming, projection, set union, selection, and set difference. The model divides embeddings into local windows and applies weighted aggregation in its scoring function. By capturing the local structure within the relations, ReAIE enables more expressive inference on n-ary databases.

3) Deep Neural Network-based Methods: Deep neural network-based position-aware models have shown remarkable adaptability in modelling n-ary facts, by using learnable components, such as CNNs, attention mechanisms, message passing frameworks, or HNNs to encode the positional and semantic representation of knowledge. Inspired by ConvE's 2D CNNs, **HyConvE** [196] uses a 3D convolutional neural network for the link prediction task. It first transforms entity and relation embeddings $\in \mathcal{R}^d$ into $\mathcal{R}^{d1\times d2}$, where $d_1 \times d_2 = d$. These 2D embeddings are then concatenated into 3D embeddings and applies position-aware convolution is applied to extract patterns between entities.

To further integrate entity contributions into relations, **LPACN** [199] combines attention mechanisms with CNNs. It constructs a relation-aware representation and encodes entity positions using convolutional kernels. Although the model is more expressive, it suffers from gradient vanishing during training. An extended version **LPACN+** is developed by adding residual connection networks to solve this problem.

HySAE [204] improves on HyCubE by introducing a more efficient 3D convolutional design to significantly reduce model parameters. Position embeddings are explicitly merged with entity embeddings. The position-enhanced 3D tensor is then passed through internal and external semantic enhancement layers, enabling multi-scale semantic learning. However, the inherent structure of CNNs is constrained by the receptive field of the convolutional kernels.

Beyond CNNs, **G-MPNN** [38] applies message passing (MP) over multi-relational ordered HG. A KHG can be viewed as a collection of directed and labelled hyperedges, i.e., a multi-relational ordered HG, which allows more flexible organisation of multi-relational facts [38], [193]. The forward propagation of an message passing neural network (MPNN) has two stages, a message passing stage and a message readout stage. At each MP layer, the model aggregates vertex features, positions, and relations within a hyperedge and updates vertex embeddings through a generalised message passing function as Eq 9:

$$m_v^{t+1} = g\left(\{M_t(h_v^t, \{(w, h_w^t)\}_{w\in e\setminus v}, R(e, P_e), P_e)\}_{e\in I_v}\right), \tag{9}$$

where M_t is the message function, R is the message readout function, P_e is the positional mapping of a hyperedge e, I_v is the set of hyperedges incident on vertex v, h_v^t, h_ω^t are the hidden representations of a vertex v and ω at time step t, and function g could be any differentiable function, e.g, element-wise mean, max or sum. While G-MPNN assumes that the relations are symmetric and is not a fully expressive model [193].

To better capture hierarchical semantics, **H²GNN** [39] encodes KHGs with hyperbolic HNNs. The model converts the KHGs into a tree-like hierarchical structure by adding n instantiated hyperedges, where n is the number of the arity. A two-stage message passing is then performed to aggregate position-aware hidden embeddings in hyperbolic space with Lorentz transformations.

B. Role-aware Models

Role-aware models explicitly encode the semantic roles that entities play in a relation. Rather than relying solely on entity position, these models assign role-specific embeddings or projections, allowing them to distinguish between entities of the same type, or the same entity in different positions with different contextual meanings. This fine-grained modelling improves relational expressiveness, especially in n-ary facts. Note that if the model encodes both positional and role information, it also falls into this category.

1) Tensor factorisation-based Models: To model n-ary relational semantics more precisely, role-aware models based on tensor factorisation methods explicitly encode role-specific embeddings for each entity and capture fine-grained entity-role compatibility. **RAM** [192] explores a latent space containing basis vectors. Entity roles are represented by linear combinations of these vectors, so that similar roles share similar semantic representations. In addition, each role is associated with a learnable pattern matrix $\mathbf{P}_i^r$, which captures the compatibility between the role r and its i_{th} entity. A multilinear scoring function is used to compute the plausibility as in Eq 10:

$$\phi(x) = \sum_{i=1}^{a_r} \left\langle \mathbf{u}_i^r, \mathbf{P}_i^r[1,:]\mathbf{E}_1, \cdots, \mathbf{P}_i^r[a_r,:]\mathbf{E}_{a_r}\right\rangle, \tag{10}$$

where $\mathbf{u}_i^r$ is the embedding vector for the i_{th} role of relation r, a_r is the arity of relation r and $\mathbf{E}_i$ maps each entity to multiple (a_r) semantic embeddings. Although RAM is fully expressive, it treats all entity-role pairs uniformly, without considering their contributions [208].

In order to address this problem, **PosKHG** [41] is a fully expressive model that encodes both positional and role information. It assigns each role n role-specific embeddings $\mathbf{c_i^r}$, where n is the number of entities. Similar to HSimplE, a position shift operator is applied to differentiate entity representations in different positions. A multilinear scoring function is as Eq 11:

$$\phi(t) = \sum_{i=1}^{\alpha} \langle \mathbf{c_r^i}, R_i^r[1,:]e_1, \ldots, R_i^r[\alpha,:]e_\alpha\rangle. \tag{11}$$

Here, $R_i^r[j,:]e_j$ denotes the compatibility between entity e_j and role ρ_i^r.

2) Deep Neural Network-based Models: Role-aware models based on deep neural networks capture the role semantics of entities in n-ary facts, typically by modelling entity-role pairs. There are three main cases: (i) facts are represented as tuples of role-entity pairs; (ii) facts are represented as HKGs, where qualifiers are sets of role-entity pairs; and (iii) models that encode each role semantic in the n-ary fact from the representation of the primary relation.

In case (i), we have **NaLP** [195], which represents n-ary facts as tuples $rel\{r_1 : v_1, \ldots, r_m : v_m\}$. CNNs are used to capture the dependencies between these role-entity pairs. NaLP is a permutation-invariant model since the order of role-entity pairs does not affect the predictions. It can support mixed-arity facts, but is not fully expressive and cannot differentiate major and minor role-entity pairs [42], [208].

In case (ii), the n-ary facts, also called hyper-relational facts (see Def 4), are modelled as HKGs, with a primary triple and a set of qualifiers. **HINGE** [201] and **NeuInfer** [34] both assume that there is a high correlation between the primary triple and its qualifiers in the valid facts. HINGE employs CNNs to capture feature vectors from primary triples and from quintuples, consisting of a triple and a qualifier pair. Then a minimum aggregateor is applied, so that the final predictions can be based mainly on the primary triple. NeuInfer uses similar ideas, but uses fully connected networks (FCNs) instead of CNNs to learn interaction features. These models capture role-entity interactions independently, do not make full use of qualifiers and ignore semantic correlations between roles [42], [46], [192].

STARE [42] incorporates an encoder-decoder architecture, using CompGCN [172] as the encoder to aggregate and update embeddings of the primary triple and qualifiers through message passing, and a Transformer-based decoder for missing entity predictions. However, STARE does not explicitly model entities in qualifiers and lacks fine-grained interaction modelling between qualifiers and the primary triple [45], [46]. **STARQE** [205] extends STARE by integrating logical query symbols ($\wedge$, $\exists$) to parameterise qualifiers, combined with a graph attention network (GAT)-based encoder for enhanced multi-hop reasoning, but it does not support literals in qualifiers [195]. To accommodate STARE's coarse qualifier modelling, **GRAN** [46] represents each hyper-relational fact as a heterogeneous graph. It introduces a Transformer-based multi-head attention mechanism with edge-biased attentional biases that captures detailed interactions between relations and entities. Nonetheless, GRAN requires a large number of parameters and does not handle numerical literals [206], [208].

For computational efficiency, **Hy-Transformer** [207] replaces the GNNs-based encoders with lightweight embedding layers (Layer Normalization + Dropout Layer). It further introduces a qualifier-masking auxiliary task, but its simplified decoder structure risks missing more subtle structural information [46]. Further addressing qualifier modelling, **QUAD** [45] uses dual aggregation encoders (one for the base triple and another for the qualifiers) alongside a Transformer-based decoder, effectively modelling bidirectional interactions. However, its complex structure increases the risk of overfitting due to the large number of model parameters [208]. Finally, **SDK** [210] extends STARE by incorporating dynamic HCNs and self-attention mechanisms. It explicitly models various qualifier interaction functions (e.g., rotation, multiplication), but relies heavily on high quality input data and carefully structured relational contexts for optimal performance [210].

Inspired by BoxE, **ShrinkE** [208] represents hyper-relational facts by explicitly modelling logical properties of qualifiers (monotonicity, implication, mutual exclusion). Entities are represented as points, and relations as spatial functions that map these points to query boxes via translation and rotation operations. Qualifiers will further "shrink" these boxes. Despite ShrinkE is lightweight, its qualifier monotonicity assumption, adding any qualifiers will not expand the set of possible answers, limits its effectiveness for opaque qualifiers [208]. To alleviate the limitations of ShrinkE, **HyperMono** [209] proposes a two-stage reasoning approach. It first performs coarse-grained triple inference, then refines the predictions with qualifiers, representing triples as cone representations that the qualifiers will progressively shrink.

To integrate global and local HG semantics, **HAHE** [211] introduces a hierarchical attention architecture. It employs dual-attention aggregation for global node-hyperedge interactions and node/edge-biased attention to capture local semantics between triples and qualifiers, achieving effective multi-position prediction. By focusing specifically on local semantics to mitigate global semantic noise in HAHE, **HyperFormer** [212] employs a Transformer-based entity-neighbour aggregation module. It learns refined qualifier embeddings by modelling bidirectional interactions between entities, relations, and qualifiers via convolution. In addition, a sparse gated Mixture-of-Experts mechanism reduces redundancy in computations, thereby improving the efficiency of the model [212].

Finally, **HyNT** [206] can efficiently handle numerical literals (time, quantity, etc.) in hyper-relational facts. It uses a context Transformer with position embeddings that distinguish between triples and qualifiers, employing specialised masks for discrete and numerical predictions.

In case (iii), role-aware modelling assumes that the main relation r implicitly encodes different role semantics. **RHKH** [198] follows this paradigm by decomposing r into a set of role-specific vectors $r_1, \ldots, r_n$, where r_i denotes the entity role at position i. For each hyperedge, it aggregates interactions between entities and their associated r_i, and applies a position shift function $\mathbf{f_r^p}$ to encode different positional semantics. The scoring function is computed by Eq 12:

$$\phi_{r_i(e_1,\ldots,e_{|r_i|})} = \frac{1}{|r_i|}\sum_{p=1}^{|r_i|} f_{r_i}^{p}(e_1,\ldots,e_{|r_i|}). \tag{12}$$

RHKH unifies position, role, and structural information, allowing to capture fine-grained representations across different relational structures.

3) Logic Rules and Hyperedge Expansion-based Models: Several papers explore alternative techniques to encode role semantics via logic rules or hyperedge expansion [40], [213], [214]. **HyperMLN** [213] is a hybrid framework that combines KHG embeddings with Markov Logic Networks (MLNs) [217]. It introduces an n-ary facts-adapted MLN that avoids the star-to-clique transformation, and captures role semantics via weighted first-order logic rules. A variational EM algorithm jointly optimises the KHG embeddings and logic rule weights: the E-step effectively distils the knowledge into KHG embeddings; the M-step updates the rule weights and the model parameters. HyperMLN achieves both predictive performance and interpretability, but is sensitive to the coverage and quality of the mined logic rules [213].

TransEQ [214] is the first fully expressive model using the equivalent transformation. It transforms HKGs into KGs by introducing a mediator entity and sub-relations, while fully preserving the original semantics and structure of HKGs. The model adopts an encoder-decoder framework, where the GNN-based encoder learns structural representations and the decoder captures semantics information. Its scoring function is as Eq 13:

$$\phi(x) = \langle \psi(h_r^L, h_{a_1}^L, \ldots, h_{a_n}^L), h_s^L, h_o^L, h_{v_1}^L, \ldots, h_{v_n}^L \rangle, \tag{13}$$

where L is the number of GNN layers, ψ denotes the aggregation function, and $\langle\cdot\rangle$ is the multilinear product.

HyperMLN highlights the potential of logic rule-based methods to improve model interpretability. TransEQ, on the other hand, demonstrates the effectiveness of the hyperedge expansion method in learning complex higher-order and hierarchical structures.

C. Aware-less Models

Aware-less models treat the set of entities in an n-ary relation without explicitly modelling their positions or semantic roles. These models assume that all participating entities contribute equally and that their positions do not affect the representation of the n-ary fact. Any ordering present is typically incidental to the construction of the dataset rather than the explicit encoding of the entity positions during the RL. This category offers simplicity and efficiency, but may lack the ability to capture fine-grained relational semantics.

1) Translation-based Methods: Translation-based aware-less models encode entities and relations through geometric translations in the embedding space. **RAE** [190] extends the translation-based paradigm to n-ary relations by modelling $r(x_1, x_2, \ldots, x_m)$ as a fixed-order tuple. It builds on the embedding loss of m-TransH [37] and introduces an additional entity relatedness loss that promotes connections between entity pairs within a fact. Although entity order is preserved in the datasets, position and role information is not explicitly modelled. This design improves link prediction performance over m-TransH, but remains limited in capturing fine-grained semantics due to position and role ignorance [193]–[195].

2) Tensor Factorisation-based Models: Tensor factorisation-based awareless methods represent an n-ary relational fact as a higher-order tensor and factorise it into structured lower-rank components. The simplest way to encode n-ary facts is to extend the existing KGRL models. **r-SimplE** [33] adapts SimplE to the n-ary setting by reification approach. For example, it decomposes a higher-order fact $r(e_1, e_2, e_3)$ into binary triples $r_1(e', e_1)$, $r_2(e', e_2)$, and $r_3(e', e_3)$, where e' is an auxiliary entity. However, the auxiliary entities are often sparse and lack rich context, leading to low-quality embeddings [33].

m-DistMult [33] chooses to extend DistMult [151] to the n-ary setting by generalising its bilinear scoring function using element-wise multiplication:

$$\phi(r(e_i, \ldots, e_j)) = \odot(r, e_i, \ldots, e_j), \tag{14}$$

where $\odot$ denotes the Hadamard product. The model is efficient and simple, but inherits some of the limitations of DistMult: it can only handle symmetric relations, and has a reduced expressiveness.

GETD [47] is the first tensor factorisation-based model designed directly for n-ary facts. It extends TuckER to n-TuckER and then applies Tensor Ring (TR) decomposition to factorize the higher-order core tensor into a circular sequence of third-order latent tensors. The scoring function is given by:

$$\phi(r_i, e_1, \ldots, e_n) = \mathrm{TR}(Z_1, \ldots, Z_k) \times_1 r_i \times_2 e_1 \cdots \times_{n+1} e_n, \tag{15}$$

where $\times_n$ is the n-mode product. GETD achieves expressiveness with linear complexity, but only supports fixed-arity facts and is sensitive to decomposition settings [42], [46], [195], [196], [213]. **S2S** [194] addresses sparsity and over-parameterisation problems of GETD. It partitions entity and relation embeddings into M segments. To reduce model parameter growth, it uses three diagonal operators (positive, negative, and neutral) to construct a sparse core tensor, resulting in a complexity of $O(m^{(n+1)})$, where $m, n \ll d_{feature}$.

Despite their efficiency and solid theoretical foundation, aware-less models based on tensor factorisation have several shared limitations in modelling n-ary relational facts. Most of these models only support fixed-arity inputs, limiting their applicability to mixed-arity facts [47]. They also lack the ability to capture asymmetric patterns or role-aware semantics, reducing their expressive power [33]. Furthermore, these models tend to be sensitive to factorisation strategies and decomposition configurations, resulting in poor generalisability across datasets [42], [46], [195], [196], [213]. While their simplicity facilitates adaptation of KGRL models, future research should address these structural and semantic limitations.

3) Deep Neural Network-based Methods: Deep neural network-based models learn n-ary relational representations through trainable architectures such as HNNs, GNNs, Transformer, and attention mechanisms.

ZHANG et al. [197] integrate BERT and hypergraph convolutional networks (HCN) to model vehicle fault knowledge base. BERT encodes textual descriptions, while HCN combines semantic and structural features of entities. Similarity is computed via a weighted sum of content-based and structure-based measures. **KHG-Aclair** [200] builds on the idea of structure-aware modelling, and introduces dual HG attention mechanism to capture higher-order interactions between nodes and relations. It also incorporates contrastive learning from the KGCL [218] to enhance embedding robustness, and applies PCA [219] for dimensionality reduction. The attention mechanism is only applied to aggregate neighbourhood information, but not for encoding importance of entities in each n-ary fact. **HyCubE** [202] further strengthens interaction modelling by introducing an end-to-end 3D convolutional architecture. It stacks 2D entity and relation embeddings into 3D tensors, introduces a novel circular convolution, and extracts salient features via max-pooling technique. This design improves both expressiveness and training efficiency. **HyperQuery** [203] departs from convolutional modelling and instead focuses on a self-supervised embedding method, so that it is also adapted to the inductive learning paradigm, which means that HyperQuery can predict and encode for the nodes that are unseened during training. It uses multilevel clustering to initialise embeddings and a novel edge2edge convolution strategy to aggregate neighbourhood information.

These models reflect a gradual evolution from early semantic-structural integration, to enhanced modelling of higher-order interactions, and finally to alternative self-supervised structural encoding strategies.

D. Temporal Extension

Recent developments in temporal knowledge representation have extended static n-ary fact modelling to time-evolving or dynamic modelling. Temporal extensions to HTKGs (see Def 5) and N-TKGs (see Def 6) aim to capture temporal dependencies and entity dynamics over time. While some models [44] integrate time into the primary quadruples in hyper-relational facts, others [43] treat temporality via a set of timestamped graph snapshots.

HyperTKG [44] extends static hyper-relational modelling by introducing a temporal quadruple (h, r, t, T), where T is a timestamp. It employs a Qualifier-Attentional Time-Aware Graph Encoder (QATGE) that aggregates entity representations from temporal neighbours through attention mechanisms.

A Transformer-based Qualifier Matching Decoder (QMD) is then used to predict missing entities. Currently, HyperTKG only predicts missing entities from the main quadruples and does not support predictions in qualifiers or timestamps. [44].

NE-Net [43] represents temporal facts as sequences of timestamped KGs, modelling each fact as $(predicate, \rho_1 : e_1, \ldots, \rho_n : e_n, t)$. NE-Net jointly learns entity and relation evolutions using two encoders. A CompGCN-based [172] encoder models the overall entity-relation interactions from all concurrent facts at each timestamp, and am RGCN-based [170] encoder captures the interactions between core entities and relations. A Transformer based decoder is finally used to predict missing components in relational reasoning and entity reasoning under future timestamps.

Despite recent progress, temporal extensions for n-ary relational models remain under-explored. Future work may focus on predicting timestamps and qualifier information, incorporating fine-grained temporal dynamics, and improving scalability for real-world temporal n-ary facts.

E. Guidelines for N-ary Representation Models

Designing effective representation learning models for n-ary relational data involves multiple dimensions. Based on our two-dimensional analysis and model taxonomy, we first summarise the desirable properties of a high-quality n-ary representation learning model as follows:

- **Position and Role Awareness.** The model should consider both entity position and semantic roles in n-ary facts. It should be able to capture entity-role correlations and the compatibility between entities, roles, and relations [192], [195], [201].
- **Full Expressiveness.** A good model should represent a wide range of relational types, including symmetric, antisymmetric (one-to-many, many-to-one, many-to-many), inverse relations [33], [41], [47], [191], [193], [214].
- **Relational Algebra Support.** Models should support basic relational algebra operations (e.g., projection, selection, union, renaming) that allow high-level abstraction and structured reasoning in KHGs. A fully expressive model does not mean that it can support relational algebra operations. [193]
- **Support for Mixed-Arity Facts.** The model should be robust to datasets with fixed and mixed arity and adaptable to different relational formats [194].
- **Efficiency and Scalability.** Models need to scale to large knowledge bases with linear time and space complexity [207].
- **Interpretability.** The reasoning results should be interpretable, ideally through transparent mechanisms such as logic-rule guided models (e.g., HyperMLN), although such mechanisms are underexplored in the KHG setting [213].
- **Multimodal Capability.** The model should handle different types of input types including categorical entities, numerical data, textual descriptions, and temporal information [206].

In summary, a high quality n-ary relational representation learning model should strike a balance between *semantic expressiveness*, *computational efficiency*, and *interpretability*, while being generalisable to dynamic, multimodal, and structurally heterogeneous knowledge sources.

VI. Benchmark Datasets and Training Settings

In this section, we present the benchmark datasets that are commonly used to evaluate n-ary relational representation learning models, including both KHG and HKG structures. In addition to the dataset statistics and structure, we also discuss negative sampling strategies, which play a crucial role in the training of link prediction models. Although negative sampling is a training technique, we include it here to provide a comprehensive view of how benchmark datasets are used and extended during model learning.

A. Benchmark Datasets

Most existing KGs use the triple format (s, r, o) to represent facts, following the RDF standard [220]. However, in real-world scenarios, many factual statements have more than two entities and roles. For example, the statement "Einstein received a master's degree in mathematics at ETH Zurich" involves four roles and entities.

Studies show that in Freebase [12], 61% of relations are non-binary [33], and over one-third of entities participate in n-ary relations [37]. This phenomenon is also present in large-scale knowledge base such as Wikidata [13] and YAGO [221]. Despite this, FB15K [222] was constructed by applying the star-to-clique (S2C) transformation to convert the n-ary data into multiple triples. This transformation is irreversible and introduces a loss of information [37].

These observations highlight the need for dedicated benchmarks that learn and preserve the n-ary structures of knowledge. Due to data acquisition challenges and model complexity, standard benchmarks for these contexts are scarce [193]. This section presents a comprehensive overview of n-ary relational datasets to provide benchmarks for future representation models.

1) Tuple-based Datasets: Tuple-based datasets represent facts as n-ary tuples, where $n \geq 2$. They encode complex interactions by associating multiple entities with a single relation $r(e_1, e_2, \ldots, e_n)$. Such datasets are widely used to evaluate n-ary facts modelling. Table II summarises commonly used n-ary tuple-based datasets in terms of entity and relation counts, data splits, arity distribution, and the presence of temporal information.

a) JF17K: JF17K[1] is a multi-relational dataset derived from Freebase. It removes facts that involve only one role and filters out triples containing strings, enumeration types, or numeric literals. It suffers from data leakage: about 44.5% of the test triples are equivalent to those in the training set [42]. This makes it susceptible to over-fitting in models.

b) FB-AUTO / m-FB15K: FB-AUTO and m-FB15K[2] are both constructed from Freebase. FB-AUTO selects facts related to automobiles by retaining relations prefixed with

[1] https://github.com/lijp12/SIR

[2] https://github.com/baharefatemi/HypE/tree/master/data

TABLE II
STATISTICS ON TUPLE-BASED DATASETS. #ENT: NUMBER OF ENTITIES, #REL: NUMBER OF RELATIONS; "2/3/4/5+ ARY" INDICATES THE NUMBER OF TUPLES WITH DIFFERENT ARITIES; "TIME INFO" INDICATES WHETHER TEMPORAL INFORMATION IS PRESENT.

Dataset	#Ent	#Rel	Train	Valid	Test	2-ary	3-ary	4-ary	5+ ary	Time Info	Domain
JF17K [37]	28,645	322	77,733	–	24,915	5,199	3,277	776	86,394	×	General Info
FB-AUTO [33]	3,410	8	6,778	2,255	2,180	3,786	0	215	7,212	×	Vehicles
M-FB15K [33]	10,314	71	415,375	39,348	38,797	82,247	400,027	26	11,220	×	WikiLinks
WikiPeople-3 [47]	12,270	66	20,656	2,582	2,582	–	25,820	–	–	×	Person
WikiPeople-4 [47]	9,528	50	12,150	1,519	1,519	–	–	15,188	–	×	Person
JF17K-3 [47]	11,541	104	27,645	3,454	3,455	–	34,544	–	–	×	General Info
JF17K-4 [47]	6,536	23	7,607	951	951	–	–	9,509	–	×	General Info
NWIKI [43]	17,481	22	108,397	14,370	15,591	24,971	113,387			✓	Person
NICE [43]	10,860	20	368,868	5,268	46,159	92,462	409,901			✓	News

TABLE III
STATISTICS ON QUALIFIER-BASED DATASETS. #ENT: NUMBER OF ENTITIES, #REL: NUMBER OF RELATIONS; QUAL/%: NUMBER AND RATIO OF QUALIFIER PAIRS; TIME INFO: WHETHER TEMPORAL INFORMATION IS PRESENT.

Dataset	Statements	#Ent	#Rel	Qual/%	Train	Valid	Test	Time Info	Domain
WikiPeople [195]	369,866	34,839	375	9,482 / 2.6%	294,439	37,715	37,712	×	Person
JF17K [42]	100,947	28,645	322	32,167 / 13.6%	76,379	–	24,568	×	General Info
WD50K [42]	236,507	47,156	532	46,320 / 45.9%	166,435	23,913	46,159	×	General Info
WD50K(33) [42]	102,107	38,124	475	31,866 / 31.2%	73,406	10,568	18,133	×	General Info
WD50K(66) [42]	49,167	27,347	494	31,696 / 64.5%	35,968	5,154	8,045	×	General Info
WD50K(100) [42]	31,314	18,792	279	31,314 / 100%	22,738	3,279	5,297	×	General Info
HN-WK [206]	–	93,255	357	–	–	–	–	✓	General Info
HN-YG [206]	–	36,947	57	–	–	–	–	✓	General Info
HN-FB [206]	–	76,340	316	–	–	–	–	✓	General Info
HN-FB-S [206]	–	27,334	208	–	–	–	–	✓	General Info
Wiki-hy [44]	278,102	12,782	136	26,670 / 9.59%	111,252	13,900	13,926	✓	General Info
YAGO-hy [44]	146,332	10,385	43	10,214 / 6.98%	51,193	10,973	10,977	✓	General Info
Yelp2018 [210]	936,278	47,472	51	433,496 / 46.3%	–	–	–	×	E-commerce
Amazon-book [210]	621,713	29,714	43	256,145 / 41.2%	–	–	–	×	E-commerce
MIND [210]	683,965	92,763	86	84,073 / 12.3%	–	–	–	×	News

automobile and excluding facts defined on a single entity or involving numeric/enumerated types. The m-FB15K, on the other hand, focuses on entities aligned with the Wikilinks database, aiming to better reflect real-world entity distributions and increase the complexity of link prediction.

c) WikiPeople-3 / WikiPeople-4: The WikiPeople-3, WikiPeople-4[3] datasets are extracted from Wikidata, focusing on *human* type. Due to the sparsity of facts with arity $n \geq 5$, the authors provide filtered facts containing only 3-ary (WikiPeople-3) and 4-ary (WikiPeople-4) relational facts.

d) JF17K-3 / JF17K-4: JF17K-3 and JF17K-4[4] are subsets filtered from JF17K, containing only 3-ary and 4-ary relational facts, respectively.

e) NWIKI / NICE: NWIKI and NICE are two temporal datasets designed for modelling n-ary facts. NWIKI is derived from Wikidata by extracting timestamped facts involving human-related entities and retaining only predicates with at least three role types. NICE is built from ICEWS event records from 2005 to 2014, including core participants, timestamps, and auxiliary information such as locations. Both datasets exhibit high arity and strong temporal characteristics.

2) Qualifier-based Datasets: Qualifier-based datasets represent n-ary relational facts as a combination of a primary triple and a set of qualifier pairs. Compared to n-ary tuple-based datasets, these datasets enhance relational semantics with explicit role-entity pairs, provide rich contextual knowledge and support fine-grained link prediction. We summarise popular qualifier-based datasets in the Table III , including their facts, entity and relation statistics, qualifier density, data splits, and the availability of temporal information.

a) WikiPeople: WikiPeople[5] is extracted from Wikidata and focuses on person-related entities. Although the dataset structurally supports qualifiers, most qualifier values are literals and are typically ignored in embedding models, resulting in a low qualifier ratio [42].

b) WD50K / WD50K(33/66/100): The WD50K[6] datasets are extracted from Wikidata and contain a large number of qualified hyper-relational statements. WD50K(33), WD50K(66), and WD50K(100) are sampled subsets of WD50K containing approximately 33%, 66%, and 100% of qualified based statements, respectively.

c) HN-WK / HN-YG / HN-FB / HN-FB-S: These datasets[7] are created from Wikidata, YAGO, and Freebase. They focus on qualifiers with numerical values (e.g., years, quantities, etc.) and normalise entities with consistent SI units.

d) Wiki-hy / YAGO-hy: Wiki-hy and YAGO-hy[8] are temporal hyper-relational datasets that extend traditional TKG benchmarks: Wikidata11k and YAGO1830, respectively. Both datasets are enriched with structured timestamped facts to support fine-grained temporal reasoning over hyper-relational structures.

[3] https://github.com/liuyuaa/GETD/tree/master/
[4] https://github.com/liuyuaa/GETD/tree/master/
[5] https://github.com/gsp2014/WikiPeople
[6] https://github.com/migalkin/StarE/tree/master/data/clean
[7] https://github.com/bdi-lab/HyNT
[8] https://github.com/0sidewalkenforcer0/HypeTKG

e) Yelp2018 / Amazon-book / MIND: Yelp2018, Amazon-book, and MIND are hyper-relational datasets used in in recommendation systems. According to StarE, the HKGs for these datasets are constructed by extracting entities and qualifier values from Wikidata. Yelp2018[9] is from the 2018 Yelp challenge and captures complex interactions between users and items. Amazon-book[10] is collected from Amazon book reviews, focusing on higher-order user-item relations. MIND [11] is a large-scale news recommendation dataset built from Microsoft News click logs, containing over two million user-news interactions.

B. Negative Sampling in KHGs

Current KG negative sampling strategies focus primarily on triples, which poses challenges for their extension to n-ary relational data. Popular KG sampling techniques include uniform random sampling [111], Bernoulli sampling [223], NSCaching [224], SANS [225], self-adversarial sampling [226], and GAN-based approaches (e.g., KBGAN [174], KSGAN [227], IGAN [228]). Among these, NSCaching requires separate caches for each entity position, leading to increased complexity that is not applicable to KHGs. SANS makes assumptions about the spatial structure of k-hop neighbours, which is unsuitable for the complex topology of KHGs. GAN-based methods usually require additional pre-training, which increases the training cost for KHGs. However, the self-adversarial method in RotatE, which uses the scoring function to automatically select high quality negatives, remain flexible and applicable to KHGs.

Common negative sampling methods for KHGs are usually based on uniform random sampling [229]. For each positive n-ary fact of length $|r|$, negative samples are generated by randomly replacing entities at each position independently. Specifically, for each fact, a negative sample set is generated as follows:

$$\bigcup_{i=1}^{k} N_x^{(i)} = \bigcup_{i=1}^{k} \{(e_1, \ldots, \bar{e}_i, \ldots, e_k) \mid \bar{e}_i \in \mathcal{E}, \bar{e}_i \neq e_i\}, \quad (16)$$

where $\bar{e}_i$ denotes the negative entity that replaces the original entity e_i at position i, and $\mathcal{E}$ represents the set of entities in the knowledge base. For example, given a positive fact *(**Director**:AngLEE, **Movie**:CrouchingTiger, HiddenDragon, **Year**:2000)*, uniform sampling randomly replaces "Crouching Tiger, Hidden Dragon" with other entity from the knowledge base. However, this strategy often produces negative samples with mismatched entity types or irrelevant semantics (e.g., replacing "Crouching Tiger, Hidden Dragon" with "Jackie Chan") which leads to the zero loss problem [228] during training and consequently reduces the training efficiency.

To overcome these issues, **HyperGAN** [229] employs generative adversarial networks (GANs) to produce high quality negatives without pre-training, thus overcoming the zero loss problem. It uses a generator to produce negative samples and a discriminator to distinguish real samples from generated samples.

[9]https://business.yelp.com/data/resources/open-dataset/
[10]kaggle.com/datasets/mohamedbakhet/amazon-books-reviews
[11]https://msnews.github.io/

VII. Summary & Open Challenges

In this survey, we systematically investigated representation learning methods for n-ary relational knowledge bases by proposing a two-dimensional taxonomy based on technical and model awareness perspectives.

First, we unified various definitions of n-ary relational knowledge data, including KHGs, HKGs, HTKGs, N-TKGs, link prediction task formulations and objectives, and the fully expressive models. We then introduced a comprehensive two-dimensional taxonomy that categorised existing methods from a technical perspective (e.g., tensor factorisation-based, translation-based, deep neural network-based, logic rule-based, and hyperedge expansion-based) and a model awareness perspective (aware-less, position-aware, and role-aware models). Through this progressive and detailed analysis, we have illustrated the evolution of expressiveness and reasoning capabilities between different approaches.

Furthermore, based on insights from existing models, we explicitly outlined desirable properties and requirements for an ideal n-ary relational representation learning approach, such as entity position and role awareness, relational algebraic expressiveness, interpretability, and multimodal data handling capabilities. To establish a standardised benchmark in the field, we summarised existing n-ary relational datasets and analysed negative sampling techniques commonly used in KHGs. Finally, we identified several promising future directions for the field as follows:

- **End-to-End Frameworks and Extensibility:** Traditional n-ary knowledge representation learning architectures involve repetitive and redundant feature mapping processes for entities and relations, and require a large number of model parameters. Designing end-to-end approaches with a unified feature mapping framework can significantly reduce parameter complexity and enhance interactions between elements. In addition, end-to-end frameworks naturally support mixed-arity facts. However, constructing a universally efficient and compact end-to-end embedding architecture for mixed-arity tuples remains a challenging task [202], [204].
- **Generating High Quality Negative Samples:** Existing negative sampling methods (e.g., uniform sampling) often generate low quality negative samples and suffer from the zero loss problem [228], which limits model performance. Future research could explore adversarial methods [226] or GAN-based generators [229] to generate high quality negative samples, thus improving the performance and robustness of models.
- **Dynamic and Temporal n-ary Fact Modelling:** Real-world n-ary facts evolve continuously over time. Future work could emphasise representation learning for dynamic or temporal n-ary facts, by developing models that efficiently capture and exploit temporal dynamics [43], [44].
- **Inductive Learning for Generalisation:** Current models primarily rely on transductive learning [33], [37], [42],

[47], [195], [201], where all entities and relations must appear during training, thus limiting the model's ability to generalise to new, unseen data. In contrast, inductive learning aims to encode and reason over entities or relations that are not observed during training. Developing inductive models that support such generalisation would greatly enhance the applicability of representation learning methods to evolving and incomplete knowledge bases [38].
- **Model Interpretability:** Most existing n-ary representation learning models are still considered as black-box systems, providing limited insight into decision making processes in domains such as medicine or finance. A notable attempt by Chen et al. [213] introduces interpretability by incorporating logic rules and Markov Logic Networks (MLNs) into n-ary relational reasoning. Future work should focus on designing more explicable frameworks, such as extending explicability techniques from KGs to n-ary datasets, to make inference paths traceable and model decisions more transparent and trustworthy.
- **Multimodal Information Fusion:** Real-world knowledge often involves multimodal data (e.g., textual, numerical, and visual). Developing methods for effective multimodal integration could significantly improve semantic understanding and reasoning capabilities [206].

Advances in these research directions are expected to significantly push the development and application of representation learning methods for n-ary relational knowledge.

References

[1] S. Ji, S. Pan, E. Cambria, P. Marttinen, and S. Y. Philip, "A survey on knowledge graphs: Representation, acquisition, and applications," *IEEE transactions on neural networks and learning systems*, vol. 33, no. 2, pp. 494–514, 2021.

[2] I. Aliyu, A. Kana, S. Aliyu *et al.*, "Development of knowledge graph for university courses management," *International Journal of Education and Management Engineering*, vol. 10, no. 2, p. 1, 2020.

[3] Y. Chi, Y. Qin, R. Song, and H. Xu, "Knowledge graph in smart education: A case study of entrepreneurship scientific publication management," *Sustainability*, vol. 10, no. 4, p. 995, 2018.

[4] Y. Chen, H. Sack, and M. Alam, "Analyzing social media for measuring public attitudes toward controversies and their driving factors: a case study of migration," *Soc. Netw. Anal. Min.*, vol. 12, no. 1, p. 135, 2022.

[5] F. Gong, M. Wang, H. Wang, S. Wang, and M. Liu, "Smr: medical knowledge graph embedding for safe medicine recommendation," *Big Data Research*, vol. 23, p. 100174, 2021.

[6] D. Cheng, F. Yang, S. Xiang, and J. Liu, "Financial time series forecasting with multi-modality graph neural network," *Pattern Recognition*, vol. 121, p. 108218, 2022.

[7] C. Peng, F. Xia, M. Naseriparsa, and F. Osborne, "Knowledge graphs: Opportunities and challenges," *Artificial Intelligence Review*, vol. 56, no. 11, pp. 13 071–13 102, 2023.

[8] M. Alam, G. A. Gesese, and P.-H. Paris, "Neurosymbolic methods for dynamic knowledge graphs," *arXiv preprint arXiv:2409.04572*, 2024.

[9] T. Pedersen, S. Patwardhan, J. Michelizzi *et al.*, "Wordnet:: Similarity-measuring the relatedness of concepts." in *AAAI*, vol. 4, 2004, pp. 25–29.

[10] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "Dbpedia: A nucleus for a web of open data," in *international semantic web conference*. Springer, 2007, pp. 722–735.

[11] F. M. Suchanek, M. Alam, T. Bonald, L. Chen, P. Paris, and J. Soria, "YAGO 4.5: A large and clean knowledge base with a rich taxonomy," in *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024*, G. H. Yang, H. Wang, S. Han, C. Hauff, G. Zuccon, and Y. Zhang, Eds. ACM, 2024, pp. 131–140. [Online]. Available: https://doi.org/10.1145/3626772.3657876

[12] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: a collaboratively created graph database for structuring human knowledge," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, 2008, pp. 1247–1250.

[13] D. Vrandečić and M. Krötzsch, "Wikidata: a free collaborative knowledgebase," *Communications of the ACM*, vol. 57, no. 10, pp. 78–85, 2014.

[14] L. Wang, J. Ren, B. Xu, J. Li, W. Luo, and F. Xia, "Model: Motif-based deep feature learning for link prediction," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 2, pp. 503–516, 2020.

[15] F. Akrami, M. S. Saeef, Q. Zhang, W. Hu, and C. Li, "Realistic re-evaluation of knowledge graph completion methods: An experimental study," in *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 2020, pp. 1995–2010.

[16] L. Bauer, Y. Wang, and M. Bansal, "Commonsense for generative multi-hop question answering tasks," *arXiv preprint arXiv:1809.06309*, 2018.

[17] S. Shin, X. Jin, J. Jung, and K.-H. Lee, "Predicate constraints based question answering over knowledge graph," *Information Processing & Management*, vol. 56, no. 3, pp. 445–462, 2019.

[18] A. Saxena, A. Tripathi, and P. Talukdar, "Improving multi-hop question answering over knowledge graphs using knowledge base embeddings," in *Proceedings of the 58th annual meeting of the association for computational linguistics*, 2020, pp. 4498–4507.

[19] A. Iana, M. Alam, and H. Paulheim, "A survey on knowledge-aware news recommender systems," *Semantic Web*, vol. 15, no. 1, pp. 21–82, 2024. [Online]. Available: https://doi.org/10.3233/SW-222991

[20] X. Wang, D. Wang, C. Xu, X. He, Y. Cao, and T.-S. Chua, "Explainable reasoning over knowledge graphs for recommendation," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, 2019, pp. 5329–5336.

[21] J. Liu, F. Xia, L. Wang, B. Xu, X. Kong, H. Tong, and I. King, "Shifu2: A network representation learning based model for advisor-advisee relationship mining," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 4, pp. 1763–1777, 2019.

[22] H. Tian and R. Zafarani, "Higher-order networks representation and learning: A survey," *ACM SIGKDD Explorations Newsletter*, vol. 26, no. 1, pp. 1–18, 2024.

[23] S. Fu, W. Liu, Y. Zhou, and L. Nie, "Hplapgcn: Hypergraph p-laplacian graph convolutional networks," *Neurocomputing*, vol. 362, pp. 166–174, 2019.

[24] J. Huang, C. Chen, F. Ye, W. Hu, and Z. Zheng, "Nonuniform hyper-network embedding with dual mechanism," *ACM Transactions on Information Systems (TOIS)*, vol. 38, no. 3, pp. 1–18, 2020.

[25] J. Jiang, Y. Wei, Y. Feng, J. Cao, and Y. Gao, "Dynamic hypergraph neural networks." in *IJCAI*, 2019, pp. 2635–2641.

[26] J. Payne, "Deep hyperedges: a framework for transductive and inductive learning on hypergraphs," *arXiv preprint arXiv:1910.02633*, 2019.

[27] H. T. Trung, T. Van Vinh, N. T. Tam, J. Jo, H. Yin, and N. Q. V. Hung, "Learning holistic interactions in lbsns with high-order, dynamic, and multi-role contexts," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 5, pp. 5002–5016, 2022.

[28] D. Yang, B. Qu, J. Yang, and P. Cudre-Mauroux, "Revisiting user mobility and social relationships in lbsns: a hypergraph embedding approach," in *The world wide web conference*, 2019, pp. 2147–2157.

[29] S. Maleki, D. P. Wall, and K. Pingali, "Netvec: A scalable hypergraph embedding system," in *Proc. of the Int. Conf. on Machine Learning-Workshops*, 2021.

[30] J. Shun, "Practical parallel hypergraph algorithms," in *Proceedings of the 25th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, 2020, pp. 232–249.

[31] B. Srinivasan, D. Zheng, and G. Karypis, "Learning over families of sets-hypergraph representation learning for higher order tasks," in *Proceedings of the 2021 SIAM International Conference on Data Mining (SDM)*. SIAM, 2021, pp. 756–764.

[32] C.-A. Yu, C.-L. Tai, T.-S. Chan, and Y.-H. Yang, "Modeling multi-way relations with hypergraph embedding," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 2018, pp. 1707–1710.

[33] B. Fatemi, P. Taslakian, D. Vazquez, and D. Poole, "Knowledge hypergraphs: Prediction beyond binary relations," *arXiv preprint arXiv:1906.00137*, 2019.

[34] S. Guan, X. Jin, J. Guo, Y. Wang, and X. Cheng, "Neuinfer: Knowledge inference on n-ary facts," in *Proceedings of the 58th annual meeting of the association for computational linguistics*, 2020, pp. 6141–6151.

[35] L. Cai, X. Mao, Y. Zhou, Z. Long, C. Wu, and M. Lan, "A survey on temporal knowledge graph: Representation learning and applications," *arXiv preprint arXiv:2403.04782*, 2024.
[36] J. Hoffart, F. M. Suchanek, K. Berberich, E. Lewis-Kelham, G. De Melo, and G. Weikum, "Yago2: exploring and querying world knowledge in time, space, context, and many languages," in *Proceedings of the 20th international conference companion on World wide web*, 2011, pp. 229–232.
[37] J. Wen, J. Li, Y. Mao, S. Chen, and R. Zhang, "On the representation and embedding of knowledge bases beyond binary relations," *arXiv preprint arXiv:1604.08642*, 2016.
[38] N. Yadati, "Neural message passing for multi-relational ordered and recursive hypergraphs," *Advances in Neural Information Processing Systems*, vol. 33, pp. 3275–3289, 2020.
[39] M. Li, X. Shi, C. Qiao, T. Zhang, and H. Jin, "Hyperbolic hypergraph neural networks for multi-relational knowledge hypergraph representation," *arXiv preprint arXiv:2412.12158*, 2024.
[40] A. Antelmi, G. Cordasco, M. Polato, V. Scarano, C. Spagnuolo, and D. Yang, "A survey on hypergraph representation learning," *ACM Computing Surveys*, vol. 56, no. 1, pp. 1–38, 2023.
[41] Z. Chen, X. Wang, C. Wang, and Z. Li, "Poskhg: A position-aware knowledge hypergraph model for link prediction," *Data Science and Engineering*, vol. 8, no. 2, pp. 135–145, 2023.
[42] M. Galkin, P. Trivedi, G. Maheshwari, R. Usbeck, and J. Lehmann, "Message passing for hyper-relational knowledge graphs," *arXiv preprint arXiv:2009.10847*, 2020.
[43] Z. Hou, X. Jin, Z. Li, L. Bai, S. Guan, Y. Zeng, J. Guo, and X. Cheng, "Temporal knowledge graph reasoning based on n-tuple modeling," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023, pp. 1090–1100.
[44] Z. Ding, J. Wu, J. Wu, Y. Xia, and V. Tresp, "Temporal fact reasoning over hyper-relational knowledge graphs," *arXiv preprint arXiv:2307.10219*, 2023.
[45] H. Shomer, W. Jin, J. Li, Y. Ma, and H. Liu, "Learning representations for hyper-relational knowledge graphs," in *Proceedings of the International Conference on Advances in Social Networks Analysis and Mining*, 2023, pp. 253–257.
[46] Q. Wang, H. Wang, Y. Lyu, and Y. Zhu, "Link prediction on n-ary relational facts: A graph-based approach," *arXiv preprint arXiv:2105.08476*, 2021.
[47] Y. Liu, Q. Yao, and Y. Li, "Generalizing tensor decomposition for n-ary relational knowledge bases," in *Proceedings of the web conference 2020*, 2020, pp. 1104–1114.
[48] V. Gutierrez Basulto and S. Schockaert, "From knowledge graph embedding to ontology embedding? an analysis of the compatibility between vector space representations and rules," 2018.
[49] Q. Wang, Z. Mao, B. Wang, and L. Guo, "Knowledge graph embedding: A survey of approaches and applications," *IEEE transactions on knowledge and data engineering*, vol. 29, no. 12, pp. 2724–2743, 2017.
[50] J. Cao, J. Fang, Z. Meng, and S. Liang, "Knowledge graph embedding: A survey from the perspective of representation spaces," *ACM Computing Surveys*, vol. 56, no. 6, pp. 1–42, 2024.
[51] Y. Gao, Z. Zhang, H. Lin, X. Zhao, S. Du, and C. Zou, "Hypergraph learning: Methods and practices," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 5, pp. 2548–2566, 2020.
[52] L. Zhang, J. Guo, J. Wang, J. Wang, S. Li, and C. Zhang, "Hypergraph and uncertain hypergraph representation learning theory and methods," *Mathematics*, vol. 10, no. 11, p. 1921, 2022.
[53] J. Huang, X. Liu, and Y. Song, "Hyper-path-based representation learning for hyper-networks," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2019, pp. 449–458.
[54] D. Yang, B. Qu, J. Yang, and P. Cudré-Mauroux, "Lbsn2vec++: Heterogeneous hypergraph embedding for location-based social networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, pp. 1843–1855, 2020.
[55] H. Tian, S. Jin, and R. Zafarani, "Exploiting cross-order patterns and link prediction in higher-order networks," in *2022 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2022, pp. 1–9.
[56] A. Bretto, "Hypergraph theory," *An introduction. Mathematical Engineering. Cham: Springer*, vol. 1, pp. 209–216, 2013.
[57] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.
[58] M. Chen, I. W. Tsang, M. Tan, and T. J. Cham, "A unified feature selection framework for graph embedding on high dimensional data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 6, pp. 1465–1477, 2014.
[59] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," *Advances in neural information processing systems*, vol. 18, 2005.
[60] D. Zhou, J. Huang, and B. Schölkopf, "Learning with hypergraphs: Clustering, classification, and embedding," *Advances in neural information processing systems*, vol. 19, 2006.
[61] M. Bolla, "Spectra, euclidean representations and clusterings of hypergraphs," *Discrete Mathematics*, vol. 117, no. 1-3, pp. 19–39, 1993.
[62] L. Pu and B. Faltings, "Hypergraph learning with hyperedge expansion," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2012, pp. 410–425.
[63] Y. Zhu, Z. Guan, S. Tan, H. Liu, D. Cai, and X. He, "Heterogeneous hypergraph embedding for document recommendation," *Neurocomputing*, vol. 216, pp. 150–162, 2016.
[64] J. A. Rodriguez, "On the laplacian eigenvalues and metric parameters of hypergraphs," *Linear and Multilinear Algebra*, vol. 50, no. 1, pp. 1–14, 2002.
[65] S. Saito, D. Mandic, and H. Suzuki, "Hypergraph p-laplacian: A differential geometry view," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.
[66] Y. Sun, S. Wang, Q. Liu, R. Hang, and G. Liu, "Hypergraph embedding for spatial-spectral joint feature extraction in hyperspectral images," *remote sensing*, vol. 9, no. 5, p. 506, 2017.
[67] F. Luo, B. Du, L. Zhang, L. Zhang, and D. Tao, "Feature learning using spatial-spectral hypergraph discriminant analysis for hyperspectral image," *IEEE transactions on cybernetics*, vol. 49, no. 7, pp. 2406–2419, 2018.
[68] F. Luo, T. Guo, Z. Lin, J. Ren, and X. Zhou, "Semisupervised hypergraph discriminant learning for dimensionality reduction of hyperspectral image," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 4242–4256, 2020.
[69] H. Yuan and Y. Y. Tang, "Learning with hypergraph for hyperspectral image feature extraction," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 8, pp. 1695–1699, 2015.
[70] S. Huang, D. Yang, Y. Ge, and X. Zhang, "Discriminant hyper-laplacian projections and its scalable extension for dimensionality reduction," *Neurocomputing*, vol. 173, pp. 145–153, 2016.
[71] H. Gui, J. Liu, F. Tao, M. Jiang, B. Norick, L. Kaplan, and J. Han, "Embedding learning with events in heterogeneous information networks," *IEEE transactions on knowledge and data engineering*, vol. 29, no. 11, pp. 2428–2441, 2017.
[72] H. Gui, J. Liu, F. Tao, M. Jiang, B. Norick, and J. Han, "Large-scale embedding learning in heterogeneous event data," in *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE, 2016, pp. 907–912.
[73] J. Sybrandt, R. Shaydulin, and I. Safro, "Hypergraph partitioning with embeddings," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 6, pp. 2771–2782, 2020.
[74] D. Yang, B. Qu, J. Yang, and P. Cudre-Mauroux, "Revisiting user mobility and social relationships in lbsns: a hypergraph embedding approach," in *The world wide web conference*, 2019, pp. 2147–2157.
[75] H. T. Trung, T. Van Vinh, N. T. Tam, J. Jo, H. Yin, and N. Q. V. Hung, "Learning holistic interactions in lbsns with high-order, dynamic, and multi-role contexts," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 5, pp. 5002–5016, 2022.
[76] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.
[77] J. Huang, C. Chen, F. Ye, J. Wu, Z. Zheng, and G. Ling, "Hyper2vec: Biased random walk for hyper-network embedding," in *Database Systems for Advanced Applications: DASFAA 2019 International Workshops: BDMS, BDQM, and GDMA, Chiang Mai, Thailand, April 22–25, 2019, Proceedings 24*. Springer, 2019, pp. 273–277.
[78] Y. Chu, C. Feng, and C. Guo, "Social-guided representation learning for images via deep heterogeneous hypergraph embedding," in *2018 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2018, pp. 1–6.
[79] V. La Gatta, V. Moscato, M. Pennone, M. Postiglione, and G. Sperlí, "Music recommendation via hypergraph embedding," *IEEE transactions on neural networks and learning systems*, vol. 34, no. 10, pp. 7887–7899, 2022.
[80] R. Zhang, Y. Zou, and J. Ma, "Hyper-sagnn: a self-attention based graph neural network for hypergraphs," *arXiv preprint arXiv:1911.02613*, 2019.

[81] Y. Zhu and H. Zhao, "Hypernetwork representation learning with the set constraint," *Applied Sciences*, vol. 12, no. 5, p. 2650, 2022.

[82] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 701–710.

[83] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proceedings of the 24th international conference on world wide web*, 2015, pp. 1067–1077.

[84] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.

[85] Y. Feng, H. You, Z. Zhang, R. Ji, and Y. Gao, "Hypergraph neural networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, 2019, pp. 3558–3565.

[86] N. Yadati, M. Nimishakavi, P. Yadav, V. Nitin, A. Louis, and P. Talukdar, "Hypergcn: Hypergraph convolutional networks for semi-supervised learning and combinatorial optimisation," *arXiv preprint arXiv:1809.02589*, 2019.

[87] X. Sun, H. Yin, B. Liu, H. Chen, J. Cao, Y. Shao, and N. Q. Viet Hung, "Heterogeneous hypergraph embedding for graph classification," in *Proceedings of the 14th ACM international conference on web search and data mining*, 2021, pp. 725–733.

[88] X. Wu, Q. Chen, W. Li, Y. Xiao, and B. Hu, "Adahgnn: Adaptive hypergraph neural networks for multi-label image classification," in *Proceedings of the 28th ACM international conference on multimedia*, 2020, pp. 284–293.

[89] J. Huang and J. Yang, "Unignn: a unified framework for graph and hypergraph neural networks," *arXiv preprint arXiv:2105.00956*, 2021.

[90] T. Thonet, J.-M. Renders, M. Choi, and J. Kim, "Joint personalized search and recommendation with hypergraph convolutional networks," in *European Conference on Information Retrieval*. Springer, 2022, pp. 443–456.

[91] Y. Dong, W. Sawin, and Y. Bengio, "Hnhn: Hypergraph networks with hyperedge neurons," *arXiv preprint arXiv:2006.12278*, 2020.

[92] J. Yi and J. Park, "Hypergraph convolutional recurrent neural network," in *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 2020, pp. 3366–3376.

[93] J. Jo, J. Baek, S. Lee, D. Kim, M. Kang, and S. J. Hwang, "Edge representation learning with hypergraphs," *Advances in Neural Information Processing Systems*, vol. 34, pp. 7534–7546, 2021.

[94] B. Liu, P. Zhao, F. Zhuang, X. Xian, Y. Liu, and V. S. Sheng, "Knowledge-aware hypergraph neural network for recommender systems," in *Database Systems for Advanced Applications: 26th International Conference, DASFAA 2021, Taipei, Taiwan, April 11–14, 2021, Proceedings, Part III 26*. Springer, 2021, pp. 132–147.

[95] X. Xia, H. Yin, J. Yu, Q. Wang, L. Cui, and X. Zhang, "Self-supervised hypergraph convolutional networks for session-based recommendation," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, 2021, pp. 4503–4511.

[96] J. Yu, H. Yin, J. Li, Q. Wang, N. Q. V. Hung, and X. Zhang, "Self-supervised multi-channel hypergraph convolutional network for social recommendation," in *Proceedings of the web conference 2021*, 2021, pp. 413–424.

[97] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. S. Yu, "Heterogeneous graph attention network," in *The world wide web conference*, 2019, pp. 2022–2032.

[98] K. Ding, J. Wang, J. Li, D. Li, and H. Liu, "Be more with less: Hypergraph attention networks for inductive text classification," *arXiv preprint arXiv:2011.00387*, 2020.

[99] J. Wang, K. Ding, Z. Zhu, and J. Caverlee, "Session-based recommendation with hypergraph attention networks," in *Proceedings of the 2021 SIAM international conference on data mining (SDM)*. SIAM, 2021, pp. 82–90.

[100] M. Ding, X. Lin, B. Zeng, and Y. Chai, "Hypergraph neural networks with attention mechanism for session-based recommendation," in *Journal of Physics: Conference Series*, vol. 2082. IOP Publishing, 2021, p. 012007.

[101] C. Chen, Z. Cheng, Z. Li, and M. Wang, "Hypergraph attention networks," in *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. IEEE, 2020, pp. 1560–1565.

[102] J. Han, Q. Tao, Y. Tang, and Y. Xia, "Dh-hgcn: dual homogeneity hypergraph convolutional network for multiple social recommendations," in *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, 2022, pp. 2190–2194.

[103] X. Luo, J. Peng, and J. Liang, "Directed hypergraph attention network for traffic forecasting," *IET Intelligent Transport Systems*, vol. 16, no. 1, pp. 85–98, 2022.

[104] S. Bai, F. Zhang, and P. H. Torr, "Hypergraph convolution and hypergraph attention," *Pattern Recognition*, vol. 110, p. 107637, 2021.

[105] L. Xia, C. Huang, and C. Zhang, "Self-supervised hypergraph transformer for recommender systems," in *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, 2022, pp. 2100–2109.

[106] X. Ma, W. Liu, Q. Tian, and Y. Gao, "Learning representation on optimized high-order manifold for visual classification," *IEEE Transactions on Multimedia*, vol. 24, pp. 3989–4001, 2021.

[107] L. Nong, J. Wang, J. Lin, H. Qiu, L. Zheng, and W. Zhang, "Hypergraph wavelet neural networks for 3d object classification," *Neurocomputing*, vol. 463, pp. 580–595, 2021.

[108] K. Tu, P. Cui, X. Wang, F. Wang, and W. Zhu, "Structural deep embedding for hyper-networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, 2018.

[109] H. Fan, F. Zhang, Y. Wei, Z. Li, C. Zou, Y. Gao, and Q. Dai, "Heterogeneous hypergraph variational autoencoder for link prediction," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 8, pp. 4125–4138, 2021.

[110] Y. Chu, C. Feng, C. Guo, Y. Wang, and J.-N. Hwang, "Event2vec: heterogeneous hypergraph embedding for event data," in *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2018, pp. 1022–1029.

[111] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," *Advances in neural information processing systems*, vol. 26, 2013.

[112] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 28, 2014.

[113] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 29, 2015.

[114] G. Ji, S. He, L. Xu, K. Liu, and J. Zhao, "Knowledge graph embedding via dynamic mapping matrix," in *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (volume 1: Long papers)*, 2015, pp. 687–696.

[115] H. Xiao, M. Huang, Y. Hao, and X. Zhu, "Transa: An adaptive approach for knowledge graph embedding," *arXiv preprint arXiv:1509.05490*, 2015.

[116] G. Ji, K. Liu, S. He, and J. Zhao, "Knowledge graph completion with adaptive sparse transfer matrix," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, 2016.

[117] D. Q. Nguyen, K. Sirts, L. Qu, and M. Johnson, "Stranse: a novel embedding model of entities and relationships in knowledge bases," *arXiv preprint arXiv:1606.08140*, 2016.

[118] Z. Zhang, J. Cai, Y. Zhang, and J. Wang, "Learning hierarchy-aware knowledge graph embeddings for link prediction," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, 2020, pp. 3065–3072.

[119] S. Wang, X. Wei, C. N. Dos Santos, Z. Wang, R. Nallapati, A. Arnold, and P. S. Yu, "Knowledge graph representation via hierarchical hyperbolic neural graph embedding," in *2021 IEEE International Conference on Big Data (Big Data)*. IEEE, 2021, pp. 540–549.

[120] S. Wang, R. Liu, L. Shen, and A. M. Khattak, "Stke: Temporal knowledge graph embedding in the spherical coordinate system," in *International Conference on Artificial Intelligence and Security*. Springer, 2022, pp. 292–305.

[121] T. Trouillon, J. Welbl, S. Riedel, É. Gaussier, and G. Bouchard, "Complex embeddings for simple link prediction," in *International conference on machine learning*. PMLR, 2016, pp. 2071–2080.

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

[123] S. Zhang, Y. Tay, L. Yao, and Q. Liu, "Quaternion knowledge graph embeddings," *Advances in neural information processing systems*, vol. 32, 2019.

[124] H. Gao, K. Yang, Y. Yang, R. Y. Zakari, J. W. Owusu, and K. Qin, "Quatde: Dynamic quaternion embedding for knowledge graph completion," *arXiv preprint arXiv:2105.09002*, 2021.

[125] J. Guo and S. Kok, "Bique: Biquaternionic embeddings of knowledge graphs," *arXiv preprint arXiv:2109.14401*, 2021.

[126] Z. Cao, Q. Xu, Z. Yang, X. Cao, and Q. Huang, "Dual quaternion knowledge graph embeddings," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, 2021, pp. 6894–6902.

[127] L. Gao, H. Zhu, H. H. Zhuo, and J. Xu, "Dual quaternion embeddings for link prediction," *Applied Sciences*, vol. 11, no. 12, p. 5572, 2021.

[128] X. Ge, Y.-C. Wang, B. Wang, and C. J. Kuo, "Core: A knowledge graph entity type prediction method via complex space regression and embedding," *Pattern Recognition Letters*, vol. 157, pp. 97–103, 2022.

[129] S. Wang, K. Fu, X. Sun, Z. Zhang, S. Li, and L. Jin, "Hierarchical-aware relation rotational knowledge graph embedding for link prediction," *Neurocomputing*, vol. 458, pp. 259–270, 2021.

[130] T. Le, N. Huynh, and B. Le, "Knowledge graph embedding by projection and rotation on hyperplanes for link prediction," *Applied Intelligence*, vol. 53, no. 9, pp. 10 340–10 364, 2023.

[131] H. Xiao, M. Huang, Y. Hao, and X. Zhu, "From one point to a manifold: Orbit models for knowledge graph embedding," in *Proceedings of IJCAI*, 2016, pp. 1315–1321.

[132] X. Lv, L. Hou, J. Li, and Z. Liu, "Differentiating concepts and instances for knowledge graph embedding," *arXiv preprint arXiv:1811.04588*, 2018.

[133] S. Wang, X. Wei, C. N. Nogueira dos Santos, Z. Wang, R. Nallapati, A. Arnold, B. Xiang, P. S. Yu, and I. F. Cruz, "Mixed-curvature multi-relational graph neural network for knowledge graph completion," in *Proceedings of the Web Conference 2021*. New York, NY, USA: Association for Computing Machinery, 2021.

[134] Y. Dong, X. Guo, J. Xiang, K. Liu, and Z. Tang, "Hypersphere: An embedding method for knowledge graph completion based on hypersphere," in *Knowledge Science, Engineering and Management: 14th International Conference, KSEM 2021, Tokyo, Japan, August 14–16, 2021, Proceedings, Part I*, 2021, p. 517–528.

[135] C. Gregucci, M. Nayyeri, D. Hernández, and S. Staab, "Link prediction with attention applied on multiple knowledge graph embedding models," in *Proceedings of the ACM web conference 2023*, 2023, pp. 2600–2610.

[136] I. Balazevic, C. Allen, and T. Hospedales, "Multi-relational poincaré graph embeddings," *Advances in neural information processing systems*, vol. 32, 2019.

[137] I. Chami, A. Wolf, D.-C. Juan, F. Sala, S. Ravi, and C. Ré, "Low-dimensional hyperbolic knowledge graph embeddings," *arXiv preprint arXiv:2005.00545*, 2020.

[138] Z. Pan and P. Wang, "Hyperbolic hierarchy-aware knowledge graph embedding for link prediction," in *Findings of the Association for Computational Linguistics: EMNLP 2021*, 2021, pp. 2941–2948.

[139] Z. Sun, M. Chen, W. Hu, C. Wang, J. Dai, and W. Zhang, "Knowledge association with hyperbolic knowledge graph embeddings," *arXiv preprint arXiv:2010.02162*, 2020.

[140] B. Xiong, S. Zhu, M. Nayyeri, C. Xu, S. Pan, C. Zhou, and S. Staab, "Ultrahyperbolic knowledge graph embeddings," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 2130–2139.

[141] K. Wang, Y. Liu, D. Lin, and Q. Z. Sheng, "Hyperbolic geometry is not necessary: Lightweight euclidean-based models for low-dimensional knowledge graph embeddings," *arXiv preprint arXiv:2103.14930*, 2021.

[142] W. Zheng, W. Wang, S. Zhao, and F. Qian, "Hyperbolic hierarchical knowledge graph embeddings for link prediction in low dimensions," *arXiv preprint arXiv:2204.13704*, 2022.

[143] Z. Cao, Q. Xu, Z. Yang, X. Cao, and Q. Huang, "Geometry interaction knowledge graph embeddings," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 36, 2022, pp. 5521–5529.

[144] T. Ebisu and R. Ichise, "Toruse: Knowledge graph embedding on a lie group," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, 2018.

[145] C. Xu and R. Li, "Relation embedding with dihedral group in knowledge graph," *arXiv preprint arXiv:1906.00687*, 2019.

[146] T. Yang, L. Sha, and P. Hong, "Nage: Non-abelian group embedding for knowledge graphs," in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 2020, pp. 1735–1742.

[147] J. Chai and G. Shi, "Module: Module embedding for knowledge graphs," *arXiv preprint arXiv:2203.04702*, 2022.

[148] T. Ebisu and R. Ichise, "Generalized translation-based embedding of knowledge graph," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, pp. 941–951, 2019.

[149] H. Lu and H. Hu, "Dense: An enhanced non-abelian group representation for knowledge graph embedding, corr abs/2008.04548 (2020)," *arXiv preprint arXiv:2008.04548*, 2020.

[150] S. M. Kazemi and D. Poole, "Simple embedding for link prediction in knowledge graphs," *Advances in neural information processing systems*, vol. 31, 2018.

[151] B. Yang, W.-t. Yih, X. He, J. Gao, and L. Deng, "Embedding entities and relations for learning and inference in knowledge bases," *arXiv preprint arXiv:1412.6575*, 2014.

[152] M. Nickel, V. Tresp, H.-P. Kriegel *et al.*, "A three-way model for collective learning on multi-relational data." in *Icml*, vol. 11, 2011, pp. 3 104 482–3 104 584.

[153] A. Bordes, J. Weston, R. Collobert, and Y. Bengio, "Learning structured embeddings of knowledge bases," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 25, 2011, pp. 301–306.

[154] M. Nickel, L. Rosasco, and T. Poggio, "Holographic embeddings of knowledge graphs (2016)," *URL: https://www. aaai. org/ocs/index. php/AAAI/AAAI16/paper/view/12484*, 2015.

[155] H. Liu, Y. Wu, and Y. Yang, "Analogical inference for multi-relational embeddings," in *International conference on machine learning*. PMLR, 2017, pp. 2168–2178.

[156] X. Ge, Y.-C. Wang, B. Wang, and C.-C. J. Kuo, "Knowledge graph embedding with 3d compound geometric transformations," *arXiv preprint arXiv:2304.00378*, 2023.

[157] I. Balažević, C. Allen, and T. M. Hospedales, "Tucker: Tensor factorization for knowledge graph completion," *arXiv preprint arXiv:1901.09590*, 2019.

[158] S. Amin, S. Varanasi, K. A. Dunfield, and G. Neumann, "Lowfer: Low-rank bilinear pooling for link prediction," in *International Conference on Machine Learning*. PMLR, 2020, pp. 257–268.

[159] A. Bordes, X. Glorot, J. Weston, and Y. Bengio, "A semantic matching energy function for learning with multi-relational data: Application to word-sense disambiguation," *Machine learning*, vol. 94, pp. 233–259, 2014.

[160] R. Socher, D. Chen, C. D. Manning, and A. Ng, "Reasoning with neural tensor networks for knowledge base completion," *Advances in neural information processing systems*, vol. 26, 2013.

[161] G. Zhao, Q. Feng, C. Chen, Z. Zhou, and Y. Yu, "Diagnose like a radiologist: Hybrid neuro-probabilistic reasoning for attribute-based medical image diagnosis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 7400–7416, 2021.

[162] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel, "Convolutional 2d knowledge graph embeddings," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, 2018.

[163] D. Q. Nguyen, T. D. Nguyen, D. Q. Nguyen, and D. Phung, "A novel embedding model for knowledge base completion based on convolutional neural network," *arXiv preprint arXiv:1712.02121*, 2017.

[164] I. Balažević, C. Allen, and T. M. Hospedales, "Hypernetwork knowledge graph embeddings," in *Artificial Neural Networks and Machine Learning–ICANN 2019: Workshop and Special Sessions: 28th International Conference on Artificial Neural Networks, Munich, Germany, September 17–19, 2019, Proceedings 28*. Springer, 2019, pp. 553–565.

[165] M. Gardner, P. Talukdar, J. Krishnamurthy, and T. Mitchell, "Incorporating vector space similarity in random walk inference over knowledge bases," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 397–406.

[166] A. Neelakantan, B. Roth, and A. McCallum, "Compositional vector space models for knowledge base completion," *arXiv preprint arXiv:1504.06662*, 2015.

[167] L. Guo, Z. Sun, and W. Hu, "Learning to exploit long-term relational dependencies in knowledge graphs," in *International conference on machine learning*. PMLR, 2019, pp. 2505–2514.

[168] Q. Wang, P. Huang, H. Wang, S. Dai, W. Jiang, J. Liu, Y. Lyu, Y. Zhu, and H. Wu, "Coke: Contextualized knowledge graph embedding," *arXiv preprint arXiv:1911.02168*, 2019.

[169] L. Yao, C. Mao, and Y. Luo, "Kg-bert: Bert for knowledge graph completion," *arXiv preprint arXiv:1909.03193*, 2019.

[170] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *The semantic web: 15th international conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, proceedings 15*. Springer, 2018, pp. 593–607.

[171] C. Shang, Y. Tang, J. Huang, J. Bi, X. He, and B. Zhou, "End-to-end structure-aware convolutional networks for knowledge base completion," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, 2019, pp. 3060–3067.

[172] S. Vashishth, S. Sanyal, V. Nitin, and P. Talukdar, "Composition-based multi-relational graph convolutional networks," *arXiv preprint arXiv:1911.03082*, 2019.

[173] D. Nathani, J. Chauhan, C. Sharma, and M. Kaul, "Learning attention-based embeddings for relation prediction in knowledge graphs," *arXiv preprint arXiv:1906.01195*, 2019.

[174] L. Cai and W. Y. Wang, "Kbgan: Adversarial learning for knowledge graph embeddings," *arXiv preprint arXiv:1711.04071*, 2017.

[175] Q. Wang, Z. Mao, B. Wang, and L. Guo, "Knowledge graph embedding: A survey of approaches and applications," *IEEE transactions on knowledge and data engineering*, vol. 29, no. 12, pp. 2724–2743, 2017.

[176] R. Xie, Z. Liu, J. Jia, H. Luan, and M. Sun, "Representation learning of knowledge graphs with entity descriptions," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, 2016.

[177] H. Xiao, M. Huang, L. Meng, and X. Zhu, "Ssp: semantic space projection for knowledge graph embedding with text descriptions," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, 2017.

[178] B. An, B. Chen, X. Han, and L. Sun, "Accurate text-enhanced knowledge graph representation learning," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018, pp. 745–755.

[179] S. Guo, Q. Wang, B. Wang, L. Wang, and L. Guo, "Semantically smooth knowledge graph embedding," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2015, pp. 84–94.

[180] R. Xie, Z. Liu, M. Sun *et al.*, "Representation learning of knowledge graphs with hierarchical types." in *IJCAI*, vol. 2016, 2016, pp. 2965–2971.

[181] Y. Lin, Z. Liu, and M. Sun, "Knowledge representation learning with entities, attributes and relations," *ethnicity*, vol. 1, pp. 41–52, 2016.

[182] Z. Zhang, F. Zhuang, M. Qu, F. Lin, and Q. He, "Knowledge graph embedding with hierarchical relation structure," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 3198–3207.

[183] R. Xie, Z. Liu, H. Luan, and M. Sun, "Image-embodied knowledge representation learning," *arXiv preprint arXiv:1609.07028*, 2016.

[184] X. Li, J. Feng, Y. Meng, Q. Han, F. Wu, and J. Li, "A unified mrc framework for named entity recognition," *arXiv preprint arXiv:1910.11476*, 2019.

[185] J. Messner, R. Abboud, and I. I. Ceylan, "Temporal knowledge graph completion using box embeddings," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, 2022, pp. 7779–7787.

[186] W. Wu, H. Li, H. Wang, and K. Q. Zhu, "Probase: A probabilistic taxonomy for text understanding," in *Proceedings of the 2012 ACM SIGMOD international conference on management of data*, 2012, pp. 481–492.

[187] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. Hruschka, and T. Mitchell, "Toward an architecture for never-ending language learning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 24, 2010, pp. 1306–1313.

[188] R. Speer, J. Chin, and C. Havasi, "Conceptnet 5.5: An open multilingual graph of general knowledge," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, 2017.

[189] X. Chen, M. Chen, W. Shi, Y. Sun, and C. Zaniolo, "Embedding uncertain knowledge graphs," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, 2019, pp. 3363–3370.

[190] R. Zhang, J. Li, J. Mei, and Y. Mao, "Scalable instance reconstruction in knowledge bases via relatedness affiliated embedding," in *Proceedings of the 2018 world wide web conference*, 2018, pp. 1185–1194.

[191] R. Abboud, I. Ceylan, T. Lukasiewicz, and T. Salvatori, "Boxe: A box embedding model for knowledge base completion," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9649–9661, 2020.

[192] Y. Liu, Q. Yao, and Y. Li, "Role-aware modeling for n-ary relational knowledge bases," in *Proceedings of the Web Conference 2021*, 2021, pp. 2660–2671.

[193] B. Fatemi, P. Taslakian, D. Vazquez, and D. Poole, "Knowledge hypergraph embedding meets relational algebra," *Journal of Machine Learning Research*, vol. 24, no. 105, pp. 1–34, 2023.

[194] S. Di, Q. Yao, and L. Chen, "Searching to sparsify tensor decomposition for n-ary relational data," in *Proceedings of the Web Conference 2021*, 2021, pp. 4043–4054.

[195] S. Guan, X. Jin, J. Guo, Y. Wang, and X. Cheng, "Link prediction on n-ary relational data based on relatedness evaluation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 1, pp. 672–685, 2021.

[196] C. Wang, X. Wang, Z. Li, Z. Chen, and J. Li, "Hyconve: A novel embedding model for knowledge hypergraph link prediction with convolutional neural networks," in *Proceedings of the ACM Web Conference 2023*, 2023, pp. 188–198.

[197] F. ZHANG, B. ZHOU, J. BAO, and X. LI, "Hypergraph embedding-based representation method for multi-nary relational knowledge of bridge crane faults," *Computer Integrated Manufacturing System*, vol. 30, no. 2, p. 445, 2024.

[198] Y. Wang, J. He, and H. Wang, "Rhkh: Relational hypergraph neural network for link prediction on n-ary knowledge hypergraph," in *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 8759–8767.

[199] J. Pang, H. Xu, H. Qin, X. Lin, X. Liu, and G. Wang, "Link prediction in knowledge hypergraph combining attention and convolution network," *Journal of Frontiers of Computer Science and Technology*, vol. 17, no. 11, pp. 2734–2742, 2023.

[200] H. Park, T. Lee, and K. Kim, "Khg-aclair: Knowledge hypergraph-based attention with contrastive learning for recommendations," *Journal of Computing Science and Engineering*, vol. 18, no. 3, pp. 169–180, 2024.

[201] P. Rosso, D. Yang, and P. Cudré-Mauroux, "Beyond triplets: hyper-relational knowledge graph embedding for link prediction," in *Proceedings of the web conference 2020*, 2020, pp. 1885–1896.

[202] Z. Li, X. Wang, J. Zhao, W. Guo, and J. Li, "Hycube: Efficient knowledge hypergraph 3d circular convolutional embedding," *IEEE Transactions on Knowledge and Data Engineering*, 2025.

[203] S. Maleki, J. Vekhter, and K. Pingali, "Hyperquery: Beyond binary link prediction," *arXiv preprint arXiv:2501.07731*, 2025.

[204] Z. Li, X. Wang, Z. Jun, F. Feng, Z. Chen, and J. Li, "Hysae: An efficient semantic-enhanced representation learning model for knowledge hypergraph link prediction," in *THE WEB CONFERENCE 2025*, 2025.

[205] D. Alivanistos, M. Berrendorf, M. Cochez, and M. Galkin, "Query embedding on hyper-relational knowledge graphs," *arXiv preprint arXiv:2106.08166*, 2021.

[206] C. Chung, J. Lee, and J. J. Whang, "Representation learning on hyper-relational and numeric knowledge graphs with transformers," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023, pp. 310–322.

[207] D. Yu and Y. Yang, "Improving hyper-relational knowledge graph completion," *arXiv preprint arXiv:2104.08167*, 2021.

[208] B. Xiong, M. Nayyer, S. Pan, and S. Staab, "Shrinking embeddings for hyper-relational knowledge graphs," *arXiv preprint arXiv:2306.02199*, 2023.

[209] Z. Hu, V. Gutiérrez-Basulto, Z. Xiang, R. Li, and J. Z. Pan, "Hypermono: A monotonicity-aware approach to hyper-relational knowledge representation," *arXiv preprint arXiv:2404.09848*, 2024.

[210] Y. Liu, H. Xuan, B. Li, M. Wang, T. Chen, and H. Yin, "Self-supervised dynamic hypergraph recommendation based on hyper-relational knowledge graph," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 2023, pp. 1617–1626.

[211] H. Luo, Y. Yang, Y. Guo, M. Sun, T. Yao, Z. Tang, K. Wan, M. Song, W. Lin *et al.*, "Hahe: hierarchical attention for hyper-relational knowledge graphs in global and local level," *arXiv preprint arXiv:2305.06588*, 2023.

[212] Z. Hu, V. Gutiérrez-Basulto, Z. Xiang, R. Li, and J. Z. Pan, "Hyperformer: Enhancing entity and relation interaction for hyper-relational knowledge graph completion," in *Proceedings of the 32nd ACM international conference on information and knowledge management*, 2023, pp. 803–812.

[213] Z. Chen, X. Wang, C. Wang, and J. Li, "Explainable link prediction in knowledge hypergraphs," in *Proceedings of the 31st ACM international conference on information & knowledge management*, 2022, pp. 262–271.

[214] Y. Liu, S. Yang, J. Ding, Q. Yao, and Y. Li, "Generalizing hyperedge expansion for hyper-relational knowledge graph modeling," *arXiv preprint arXiv:2411.06191*, 2024.

[215] F. L. Hitchcock, "The expression of a tensor or a polyadic as a sum of products," *Journal of Mathematics and Physics*, vol. 6, no. 1-4, pp. 164–189, 1927.

[216] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner *et al.*, "Relational inductive biases, deep learning, and graph networks," *arXiv preprint arXiv:1806.01261*, 2018.

[217] M. Richardson and P. Domingos, "Markov logic networks," *Machine learning*, vol. 62, pp. 107–136, 2006.

[218] Y. Yang, C. Huang, L. Xia, and C. Li, "Knowledge graph contrastive learning for recommendation," in *Proceedings of the 45th international*

ACM SIGIR conference on research and development in information retrieval, 2022, pp. 1434–1443.

[219] M. B. Cohen, S. Elder, C. Musco, C. Musco, and M. Persu, "Dimensionality reduction for k-means clustering and low rank approximation," in *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, 2015, pp. 163–172.

[220] M. Nickel, K. Murphy, V. Tresp, and E. Gabrilovich, "A review of relational machine learning for knowledge graphs," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 11–33, 2015.

[221] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: a core of semantic knowledge," in *Proceedings of the 16th international conference on World Wide Web*, 2007, pp. 697–706.

[222] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," *Advances in neural information processing systems*, vol. 26, 2013.

[223] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 28, 2014.

[224] Y. Zhang, Q. Yao, Y. Shao, and L. Chen, "Nscaching: simple and efficient negative sampling for knowledge graph embedding," in *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 2019, pp. 614–625.

[225] K. Ahrabian, A. Feizi, Y. Salehi, W. L. Hamilton, and A. J. Bose, "Structure aware negative sampling in knowledge graphs," *arXiv preprint arXiv:2009.11355*, 2020.

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

[227] K. Hu, H. Liu, and T. Hao, "A knowledge selective adversarial network for link prediction in knowledge graph," in *Natural Language Processing and Chinese Computing: 8th CCF International Conference, NLPCC 2019, Dunhuang, China, October 9–14, 2019, Proceedings, Part I 8*. Springer, 2019, pp. 171–183.

[228] P. Wang, S. Li, and R. Pan, "Incorporating gan for negative sampling in knowledge representation learning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, 2018.

[229] Z. Guo, J. Zuo, L. Duan, R. Li, C. He, Y. Xiao, and P. Wang, "A generative adversarial negative sampling method for knowledge hypergraph link prediction," *Journal of Computer Research and Development*, vol. 59, no. 8, pp. 1742–1756, 2022.