The Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI-23)

DHGE: Dual-View Hyper-Relational Knowledge Graph Embedding for Link Prediction and Entity Typing

Haoran Luo, Haihong E*, **Ling Tan, Gengxian Zhou, Tianyu Yao, Kaiyang Wan**

School of Computer Science, Beijing University of Posts and Telecommunications, Beijing, China
{luohaoran, ehaihong}@bupt.edu.cn

Abstract

In the field of representation learning on knowledge graphs (KGs), a hyper-relational fact consists of a main triple and several auxiliary attribute-value descriptions, which is considered more comprehensive and specific than a triple-based fact. However, currently available hyper-relational KG embedding methods in a single view are limited in application because they weaken the hierarchical structure that represents the affiliation between entities. To overcome this limitation, we propose a dual-view hyper-relational KG structure (DH-KG) that contains a hyper-relational instance view for entities and a hyper-relational ontology view for concepts that are abstracted hierarchically from the entities. This paper defines link prediction and entity typing tasks on DH-KG for the first time and constructs two DH-KG datasets, JW44K-6K, extracted from Wikidata, and HTDM based on medical data. Furthermore, we propose DHGE, a DH-KG embedding model based on GRAN encoders, HGNNs, and joint learning. DHGE outperforms baseline models on DH-KG, according to experimental results. Finally, we provide an example of how this technology can be used to treat hypertension. Our model and new datasets are publicly available.

1 Introduction

Modern large-scale knowledge graphs (KGs), such as Freebase (Bollacker et al. 2008) and Wikidata (Vrandecic and Krötzsch 2014), consist of facts with entities and relations. A triple-based fact has two entities and one relation, which is usually expressed by triples (*subject*, *relation*, *object*).

Recent studies (Wen et al. 2016; Guan et al. 2021) suggest that some facts contain more than two entities, which are represented as hyper-relational facts (Rosso, Yang, and Cudré-Mauroux 2020). In contrast to triples, hyper-relational facts consist of one main triple (s,r,o) and several auxiliary attribute-value descriptions $\{(a_i : v_i)\}$. For example, the fact that *Marie Curie received Nobel Prize in Physics in 1903 together with Pierre Curie and Antoine Henri Becquerel*, can be expressed as a hyper-relational fact: (*Marie Curie*, *receive*, *Nobel Prize in Physics*, *year*, *1903*, *together_with*, *Pierre Curie*, *together_with*, *Antoine Henri Becquerel*) as shown in Figure 1.

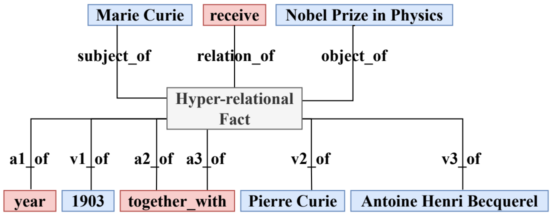

Figure 1: Hyper-relational Structure.

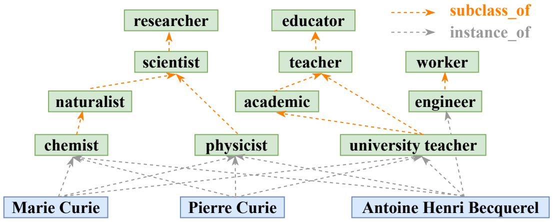

Figure 2: Hierarchical Structure.

However, the hyper-relational structure weakens the hierarchical structure of KGs due to the consideration of auxiliary information. As pointed out by Zhou et al. (2021), the KG hierarchical structure consists of triple-based facts with hierarchical relations such as "instance_of" and "subclass_of". As shown in Figure 2, (*Marie Curie*, instance_of, *chemist*) represents the affiliation between the entity *Marie Curie* and the concept *chemist*, and (*chemist*, subclass_of, *naturalist*) represents the affiliation between the concept *chemist* and the more abstract concept *naturalist*.

In real-world KGs, hierarchical and hyper relations are essential in describing knowledge facts. Despite this, no research has been conducted on the joint modeling of the two relations, severely limiting the application of the hyper-relational KG embedding methods. In our work, we are primarily concerned with overcoming this limitation.

For the first time, this paper defines the structure of hyper-relational dual-view KG (DH-KG), as Figure 3 shows, consisting of a hyper-relational instance view for entities and a hyper-relational ontology view for concepts abstracted hi-

[*]Corresponding author.

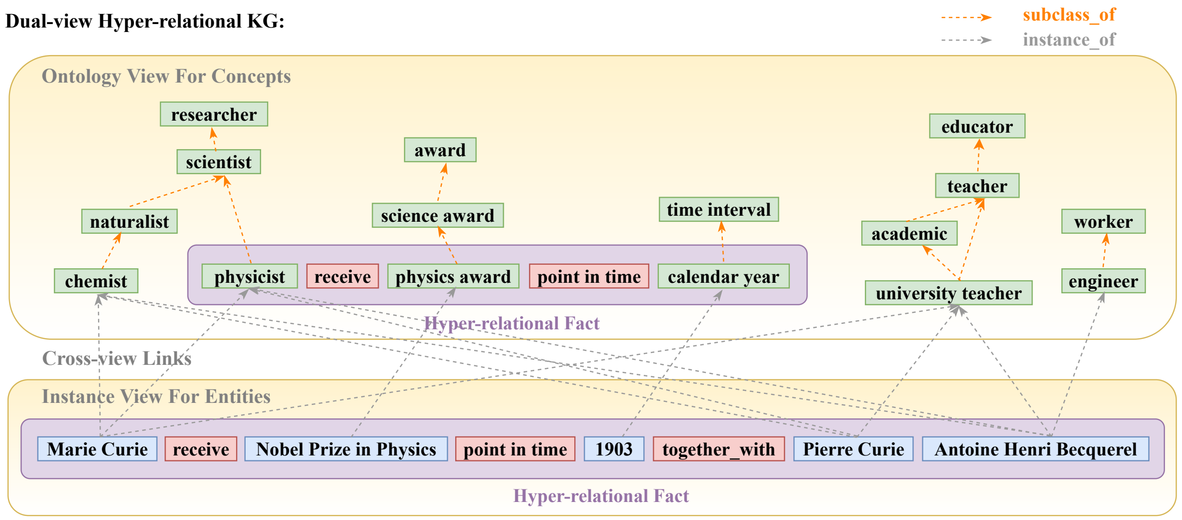

Figure 3: An example on Dual-view Hyper-Relational KG Structure, where a hyper-relational fact in the instance view consists of a main triple (*Marie Curie*, *receive*, *Nobel Prize in Physics*) and auxiliary attribute-value descriptions (*year*: *1903*, *together_with*: *Pierre Curie*, *together_with*: *Antoine Henri Becquerel*). Additionally, entities in instance view and concepts in ontology view are linked across views through the "instance_of" relation, and many concepts in the ontology view are connected by the "subclass_of" relation.

erarchically from entities. The two views are connected by cross-view links, which link entities and corresponding concepts. Based on this structure, we construct two DH-KG datasets, JW44K-6K extracted from Wikidata (Vrandecic and Krötzsch 2014), and HTDM based on medical data about hypertension and describe their construction method.

We also propose DHGE, the first model that can jointly model both hyper-relational facts and dual-view hierarchical information. DHGE utilizes the GRAN encoder (Wang et al. 2021), hypergraph neural network (Feng et al. 2019), and joint learning to perform the hyper-relational KG link prediction task within two views and the cross-view linked entity typing task. We conduct extensive experiments comparing our model with other models and analyze the results. Our model performs well on JW44K-6K. Additionally, we provide an example of HTDM application in hypertension.

Our main contributions are summarized as follows:

- We define the structure of DH-KG and its link prediction and entity typing tasks for the first time.
- We construct two DH-KG datasets, JW44K-6K and HTDM.
- We propose DHGE, the first DH-KG embedding model that can simultaneously model hyper-relational facts and hierarchical information.
- We conduct extensive experiments and comparative analyses to verify our model's good performance on DH-KGs.
- We provide an example of the application in the field of hypertension on HTDM.

2 Related Work

2.1 Knowledge Graph Datasets

Knowledge graph (KG) datasets are generally subgraphs extracted from large KGs, which can be summarized in the following four forms.

Single-view Triple-based KG Datasets. The original KG datasets are in the form of single-view triples such as FB15K, FB15K-237 (Bordes et al. 2013) from Freebase, and WN18, WN18RR (Bordes et al. 2013) from Wordnet.

Single-view Hyper-relational KG Datasets. To represent the hyper-relational facts, JF17K (Wen et al. 2016), WikiPeople (Guan et al. 2021), WD50K (Galkin et al. 2020), and other datasets extend the triplet paradigm to a hyper-relational paradigm but still in a single view.

Dual-view Triple-based KG Datasets. To better consider the hierarchical information between entities and concepts, YAGO26K-906, DB111K-174 (Hao et al. 2019), and KACC (Zhou et al. 2021) adopt the dual-view structure, but they are still limited to the triple form.

Dual-view Hyper-relational KG Datasets. Our proposed DH-KG datasets, JW44K-6K and HTDM, contain high-quality instance and ontology views with hyper-relational facts and are rich in cross-view links. To our best knowledge, JW44K-6K is currently the only DH-KG dataset closest to real-world facts, while HTDM is the only DH-KG dataset for medical applications.

2.2 Knowledge Graph Embedding Methods

The knowledge graph embedding (KGE) methods determine the performance of the knowledge representation, which can be summarized in the following four categories.

Category	Dataset	The Instance View					The Ontology View					# Cross-links
		# Entities	# Rels	# Facts	Arity	#[A>2]Facts(%)	# Concepts	# Rels	# Facts	Arity	#[A>2]Facts(%)	
ST-KGs	FB15K	14,951	1,345	582,213	2	-	-	-	-	-	-	-
	FB15K-237	14,541	237	310,116	2	-	-	-	-	-	-	-
	WN18	40,943	18	151,442	2	-	-	-	-	-	-	-
	WN18RR	40,943	11	93,003	2	-	-	-	-	-	-	-
SH-KGs	JF17K	28,645	322	100,947	2-6	46,320(45.9%)	-	-	-	-	-	-
	Wikipeople	34,839	375	369,866	2-9	9,482(2.6%)	-	-	-	-	-	-
	WD50K	47,156	532	236,507	2-7	32,167(13.6%)	-	-	-	-	-	-
DT-KGs	YAGO26K-906	26,078	34	390,738	2	-	906	30	8,962	2	-	9,962
	DB111K-174	111,762	305	863,643	2	-	174	20	763	2	-	99,748
	KACC-M	99,615	209	662,650	2	-	6,685	30	15,616	2	-	123,342
DH-KGs	JW44K-6K	44,342	630	514,822	2-10	50,869(9.9%)	6,589	141	17,667	2-7	155(0.9%)	61,874
	HTDM	1,297	45	208	7-45	208(100%)	266	19	470	2-10	296(63.0%)	469

Table 1: Data statistics of datasets of ST-KGs (Single-view Triple-based KGs), SH-KGs (Single-view Hyper-relational KGs), DT-KGs (Dual-view Triple-based KGs), and DH-KGs (Dual-view Hyper-relational KGs), where the columns indicate the number of entities, relations, facts, arity of facts, and facts with more than two arities, respectively.

Single-view Triple-based KGE Methods. Traditional KGE methods mainly model triples on a single view, such as translation models (Bordes et al. 2013; Sun et al. 2019), tensor factorization base models (Balazevic, Allen, and Hospedales 2019), and neural models (Dettmers et al. 2018).

Single-view Hyper-relational KGE Methods. Some methods, such as m-TransH (Wen et al. 2016) and NaLP (Guan et al. 2021), extend the triple-based paradigm with binary relations to multiple relations. HINGE (Rosso, Yang, and Cudré-Mauroux 2020), StarE (Galkin et al. 2020), and GRAN (Wang et al. 2021) utilize a hyper-relational paradigm to model multiple relations in a single view.

Dual-view Triple-based KGE Methods. Models such as JOIE (Hao et al. 2019) and HyperKA (Sun et al. 2020) can jointly model the information in dual-view to simultaneously model hierarchical and logical information, but they still cannot model hyper-relational facts.

Dual-view Hyper-relational KGE Methods. Our proposed model, DHGE, extends hyper-relational KG embedding to a dual-view structure. To our best knowledge, DHGE is currently the only KGE model that can jointly model dual-view hierarchical relations and hyper-relational facts.

3 Problem Statement

In this section, we present the formalization of dual-view hyper-relational KG (DH-KG) and the definition of link prediction and entity typing tasks.

Formalization of DH-KG. DH-KG contains two subviews and some cross-view links, and we denote it as $\{\mathcal{G}_I, \mathcal{G}_O, \mathcal{H}_S\}$. The instance view $\mathcal{G}_I = \{\mathcal{E}_I, \mathcal{R}_I, \mathcal{H}_I\}$ consists of an instance entity set $\mathcal{E}_I$, an instance relation set $\mathcal{R}_I$, and an instance hyper-relational fact set $\mathcal{H}_I = \{((s^I, r^I, o^I), \{(a_i^I : v_i^I)\}_{i=1}^m) | s^I, o^I, v_1^I, ..., v_m^I \in \mathcal{E}_I, r^I, a_1^I, ..., a_m^I \in \mathcal{R}_I\}$, where (s, r, o) represents the main triple and $\{(a_i^I : v_i^I)\}_{i=1}^m$ represents m auxiliary attribute-value pairs. Similarly, the ontology view $\mathcal{G}_O = \{\mathcal{C}_O, \mathcal{R}_O, \mathcal{H}_O\}$ consists of an ontology concept set $\mathcal{C}_O$, an ontology relation set $\mathcal{R}_O$, and an ontology hyper-relational fact set $\mathcal{H}_O = \{((s^O, r^O, o^O), \{(a_i^O : v_i^O)\}_{i=1}^m) | s^O, o^O, v_1^O, ..., v_m^O \in \mathcal{C}_O, r^O, a_1^O, ..., a_m^O \in \mathcal{R}_O\}$. The set of cross-view links $\mathcal{H}_S = \{(h_S, \texttt{instance\_of}, t_S)\}$ is a set of hyper-relational facts without auxiliary attribute-value pairs, where $h_S \in \mathcal{E}_I$ and $t_S \in \mathcal{C}_O$.

Link prediction (LP) on DH-KG. The hyper-relational LP task aims to predict missing elements from hyper-relation facts$((s^I, r^I, o^I), \{(a_i : v_i)\}_{i=1}^m)$. Missing elements can be entities $\in \{s, o, v_1, ..., v_m\}$ or relations $\in \{r, a_1, ..., a_m\}$. On DH-KG, $\mathcal{G}_I$ LP and $\mathcal{G}_O$ LP are hyper-relational LP in the instance and ontology view, respectively.

Entity typing (ET) on DH-KG. The ET task attempts to predict the associated concepts of some given entity. On DH-KG, the ET task is to predict the tail of $(h, \texttt{instance\_of}, t)$ in cross-view links, where $h \in \mathcal{E}_I$, $t \in \mathcal{C}_O$.

4 Dataset Construction

In this section, we introduce the construction method, data statistics, and analysis of two DH-KG datasets[1], JW44K-6K and HTDM.

4.1 JW44K-6K Dataset

JW44K-6K is a DH-KG dataset extracted from Wikidata (Vrandecic and Krötzsch 2014). It consists of 44K instance-view entities and 6K ontology-view concepts, with rich cross-view links and intra-view hyper-relational facts. The construction of the JW44K-6K dataset can be divided into four steps.

Step1. Instance-view entity filtering. A hyper-relational KG dataset WD50K, proposed by Galkin et al. 2020, is extracted from Wikidata, containing 50K entities. We take the entities of WD50K as the instance-view entity set $\mathcal{E}_I$, then

[1] https://github.com/LHRLAB/DHGE/tree/main/dataset

Splited Dataset		$\mathcal{H}_I$	$\mathcal{H}_O$	$\mathcal{H}_S$
JW44K-6K	All	514,822	17,667	61,874
	Train	412,477	14,130	49,517
	Valid	51,129	1,761	6,178
	Test	51,152	1,774	6,178
HTDM	All	208	470	469
	Train	166	376	375
	Valid	21	47	47
	Test	21	47	47

Table 2: Data Statistics for Dataset Partition of two DH-KG datasets, JW44K-6K and HTDM.

Task	Train	Valid	Test
$\mathcal{G}_I$ **LP**	$\mathcal{H}_I^{Train} \cup \mathcal{H}_O^{Train} \cup \mathcal{H}_S^{Train}$	$\mathcal{H}_I^{Valid}$	$\mathcal{H}_I^{Test}$
$\mathcal{G}_O$ **LP**	$\mathcal{H}_I^{Train} \cup \mathcal{H}_O^{Train} \cup \mathcal{H}_S^{Train}$	$\mathcal{H}_O^{Valid}$	$\mathcal{H}_O^{Test}$
ET	$\mathcal{H}_I^{Train} \cup \mathcal{H}_O^{Train} \cup \mathcal{H}_S^{Train}$	$\mathcal{H}_S^{Valid}$	$\mathcal{H}_S^{Test}$

Table 3: The usage of dataset partition in instance-view LP, ontology-view LP, and ET tasks of DH-KG.

take $\mathcal{E}_I$ as the head entity set to find the corresponding tail entity set T through the relation "instance_of" in Wikidata, and update $\mathcal{E}_I$ to $\mathcal{E}_I - \mathcal{E}_I \cap T$, thereby filtering out conceptual entities. The filtered $\mathcal{E}_I$ has 44K entities.

Step2. Ontology-view concept selection. We continue to find the tail entity set corresponding to $\mathcal{E}_I$ through the "instance_of" relation as the ontology-view concept set $\mathcal{C}_O$ on Wikidata, and we consider this "instance_of" fact set as the cross-view link set $\mathcal{H}_S$. Then, we find the tail entity set Q corresponding to $\mathcal{C}_O$ through the "subclass_of" relation and update $\mathcal{C}_O$ to $\mathcal{C}_O \cup Q$ to obtain deeper concepts of the existing concepts of the ontology view. This operation is repeated until there are no deeper concepts to finish $\mathcal{C}_O$.

Step3. Extract the hyper-relational facts from the two views, respectively. First, we extract the hyper-relational fact set $\mathcal{H}_I$, where all entities and auxiliary values in extracted facts are from $\mathcal{E}_I$. Then we extract the hyper-relation $\mathcal{H}_O$ in the ontology layer concept set, where all concepts and auxiliary values in extracted facts are from $\mathcal{C}_O$.

Step4. Dataset synthesis. We extract relation set from $\mathcal{H}_I$, $\mathcal{H}_O$ as $\mathcal{R}_I$, $\mathcal{R}_O$, and we construct the instance view $\mathcal{G}_I$=$\{\mathcal{E}_I, \mathcal{R}_I, \mathcal{H}_I\}$ and the ontology view $\mathcal{G}_O$=$\{\mathcal{E}_O, \mathcal{R}_O, \mathcal{H}_O\}$. Finally, we get a DH-KG dataset JW44K-6K, $\{\mathcal{G}_I, \mathcal{G}_O, \mathcal{H}_S\}$.

4.2 HTDM Dataset

While researching hypertension medical data, we discover that it contains many hierarchical structures suitable for DH-KG datasets. Existing medical knowledge graphs contain rich hierarchical knowledge, and patient data contains a wide range of entities. Therefore, we construct our hypertensive disease medication dataset HTDM based on DH-KG for patient medication decisions with medical knowledge by link prediction.

HTDM consists of an ontology view composed of medical knowledge and an instance view derived from actual cases, and we construct it in four steps: ontology view establishment based on medical knowledge, instance view construction based on patient data, cross-view links collection, and medicine set definition.

4.3 Dataset Analysis

Table 1 presents the data statistics of the JW44K-6K, and HTDM datasets, Table 2 depicts our division of the datasets into a training set, a validation set, and a test set in the ratio of 8:1:1, and Table 3 shows what datasets we use for LP and ET on DH-KG.

We can observe that JW44K and HTDM, as DH-KG datasets, contain hyper-relational facts and dual-view to represent the hierarchical structure and are more comprehensive than other datasets. The JW44K-6K dataset has a larger volume than HTDM, but HTDM is a medical dataset and contains hyper-relational data with greater arity.

5 DHGE Model

In this section, we introduce our proposed DH-KG embedding model DHGE[2], which consists of the intra-view GRAN encoder, cross-view association, and joint learning on DH-KG.

5.1 Intra-view GRAN Encoder

For intra-view hyper-relation learning, we use the GRAN (Wang et al. 2021) learning method to update embedding, predict the entity or relation embedding in masked position, and calculate the intra-view loss in each view.

GRAN Learning Method. GRAN learns a hyper-relational fact $\mathcal{H} = ((s, r, o), \{(a_i : v_i)\}_{i=1}^{m})$ as a heterogeneous graph, then uses the mask learning strategy to construct the input of the model. The mask strategy generates (2m+3) times as many hyper-relational facts by replacing entities at each location with [MASK]. We divide these hyper-relational facts into multiple batches for training, denoting batch_size as B. Next, GRAN uses the edge-biased fully-connected attention to learn the heterogeneous graph. We randomly embed the intra-view entity set $\mathcal{V}$ into $\mathbf{X} = (\mathbf{x_1}, \ldots, \mathbf{x}_{|\mathcal{V}|}) \in R^{|\mathcal{V}| \times d}$, where d is the entity embedding dimension, and replace entities in all hyper-relational facts with corresponding embeddings. Taking a batch $\mathbf{S} = (\mathbf{S_1}, \ldots, \mathbf{S_B}) \in R^{B \times L \times d}$ training as an example, where L is the maximum number of entities in a hyper-relational fact. We take S as the input to the GRAN encoder, and the updated S after l-th layer GRAN encoder ($GRAN\_E$) is:

$$\mathbf{S}^{(l)} = GRAN\_E(\mathbf{S}^{(l-1)}), \tag{1}$$

where $\mathbf{S}^{(l)} \in R^{B \times L \times d}$ is the output result of the l-th layer $l = 1, ..., L$.

Intra-view Loss. We utilize an L-layer GRAN encoder for node embedding updates at the instance and ontology layers. After getting the updated nodes, we take out the node

[2] https://github.com/LHRLAB/DHGE

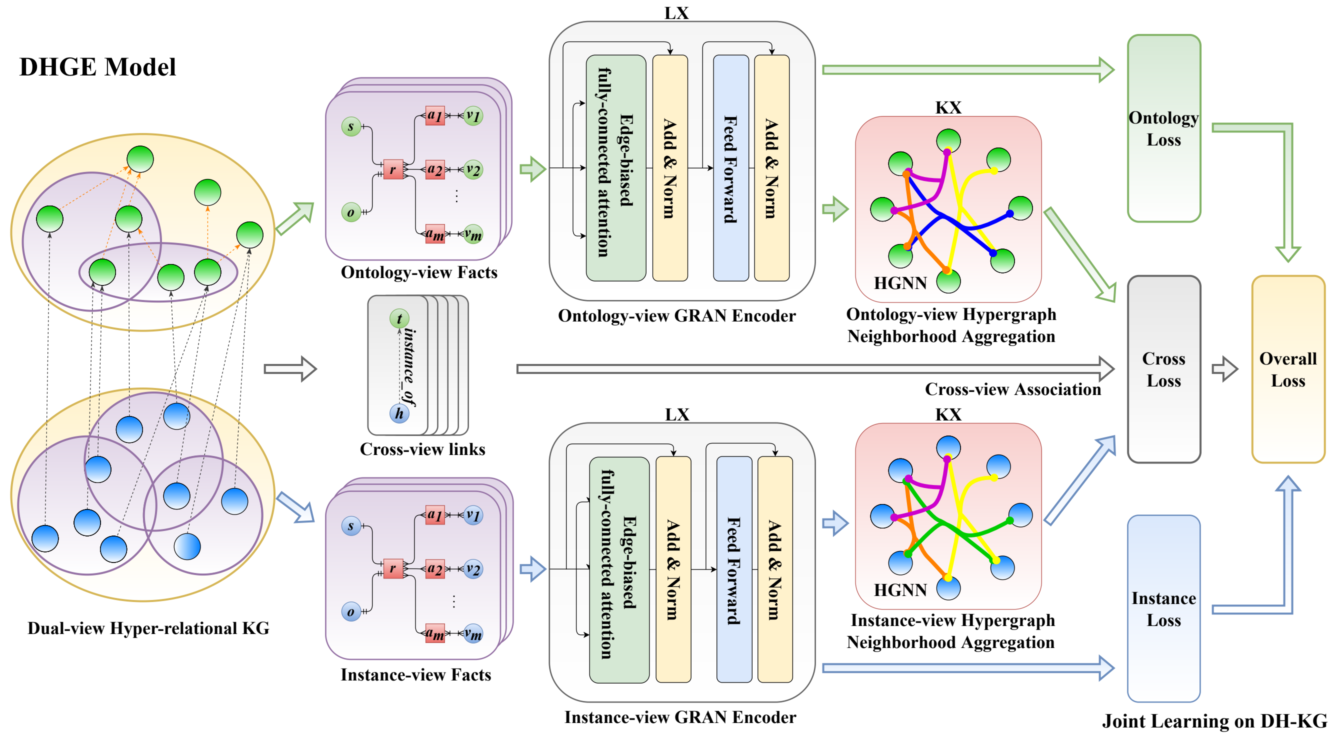

Figure 4: The DHGE model consists of three modules: GRAN encoders, HGNNs, and joint learning.

embedding matrix $\mathbf{h} \in R^{B \times d}$ corresponding to [MASK] in S, and perform an MLP and a linear layer operation,

$$\mathbf{p}_i = MLP\left(\mathbf{h}_i\right)\mathbf{W}^T, \tag{2}$$

where $\mathbf{W} \in R^{|\mathcal{V}| \times d}$ share the initial entity embedding matrix, MLP is $R^d \rightarrow R^d$, $\mathbf{h}_i$ and $\mathbf{p}_i \in R^{B \times |\mathcal{V}|}$ are the i-th item of $\mathbf{h}$ and $\mathbf{p}$, respectively, where $\mathbf{p}$ is the similarity matrix of all [MASK] embeddings in S with all entity embeddings. Finally, the cross-entropy with label smoothing between the prediction and the label is calculated as the intra-view training loss:

$$\mathcal{L}_{intra} = \sum_{t=1}^{B} \mathbf{y}_t \log \frac{exp\left(\mathbf{p}_t\right)}{\sum_{k=1}^{|V|} exp\left(\mathbf{p}_k\right)}, \tag{3}$$

where $\mathbf{y}_t$ is the t-th entry of the label $\mathbf{y}$.

Therefore, the loss obtained at the instance layer is recorded as $\mathcal{L}_{intra}^{\mathcal{G}_I}$, and the loss obtained at the ontology layer is recorded as $\mathcal{L}_{intra}^{\mathcal{G}_O}$.

5.2 Cross-view Association

To complete the cross-view association, we propose using hypergraph neighborhood aggregation and cross-view loss function after learning the intra-view GRAN encoder.

Hypergraph Neighborhood Aggregation. Because of two or more arity hyper-relational facts, each view of DH-KG can be regarded as a hypergraph $\mathcal{G}_H = (\mathcal{V}, \mathcal{E})$ composed of entity nodes and hyper-edges. To combine the information of the two views, we use HGNN (Feng et al. 2019) to aggregate the node information with hyper-edges. Through the GRAN encoder used before, we have obtained all node embedding $\mathbf{U}^{(0)} = (\mathbf{u}_1, ..., \mathbf{u}_n) = \mathbf{X}^{(L)} \in R^{|\mathcal{V}| \times d}$ as the input of HGNN. The message passing process of the HGNN from layer $(k-1)$ to layer k is defined as follows:

$$\mathbf{W}_H = \mathbf{D}_v^{-1/2}\mathbf{H}\mathbf{D}_e^{-1}\mathbf{H}^{\top}\mathbf{D}_v^{-1/2}, \tag{4}$$

$$\mathbf{U}^{(k)} = \mathbf{U}^{(k-1)} + \sigma\left(\mathbf{W}_H\mathbf{U}^{(k-1)}\mathbf{\Theta}^{(k)} + \mathbf{b}^{(k)}\right), \tag{5}$$

where $\mathbf{\Theta}^{(k)} \in R^{|\mathcal{V}| \times |\mathcal{V}|}$ is the transformation matrix, $\mathbf{b}^{(k)} \in R^{|\mathcal{V}|}$ is the bias vector of the kth layer, σ is an activation function, $\mathbf{H} \in R^{|\mathcal{V}| \times |\mathcal{E}|}$ is the association matrix of the knowledge hypergraph, $\mathbf{D}_v \in R^{|\mathcal{V}|}$ is the degree of the node, $\mathbf{D}_e \in R^{|\mathcal{E}|}$ is the degree of the hyperedge, and $\mathbf{U}^{(k)} \in R^d$ is the output result of the k-th layer $k = 1, ..., K$. We combine the input and output representations $\mathbf{U} = \mathbf{U}^{(0)} + \mathbf{U}^{(K)}$ as the final embedding to further benefit from relational transformation

Cross-view Loss. We obtain $\mathbf{U}^{\mathcal{G}_I}$ and $\mathbf{U}^{\mathcal{G}_O}$ after Hypergraph Neighborhood Aggregation on the instance layer and the ontology layer, respectively. According to the cross-view link set H_S, the embedding of the head entity h_S is in the embedding space of the instance view denoted as $\mathbf{U}^{\mathcal{G}_I}(h_S)$, and the embedding of the tail entity t_S is in the ontology view embedding space denoted as $\mathbf{U}^{\mathcal{G}_O}(t_S)$. We map the head entity set to the same ontology view embedding space as the tail entity through the mapping operation:

$$f(\mathbf{U}^{\mathcal{G}_I}(h_S)) = \mathbf{W}_{map}(\mathbf{U}^{\mathcal{G}_I}(h_S)) + \mathbf{b}_{map}, \tag{6}$$

Catagory	Model	JW44K-6K: $\mathcal{G}_I$ Link Prediction								JW44K-6K: $\mathcal{G}_O$ Link Prediction							
		Entity Prediction				Relation Prediction				Entity Prediction				Relation Prediction			
		MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
ST-KGE	TransE	0.197	0.091	0.248	0.404	-	-	-	-	0.094	0.027	0.135	0.255	-	-	-	-
	RotatE	0.243	0.152	0.280	0.425	-	-	-	-	0.162	0.051	0.209	0.298	-	-	-	-
	TuckER	0.296	0.221	0.330	0.435	-	-	-	-	0.182	0.077	0.161	0.278	-	-	-	-
	AttH	0.245	0.170	0.271	0.384	-	-	-	-	0.137	0.056	0.159	0.309	-	-	-	-
SH-KGE	HINGE	0.227	0.161	0.255	0.354	0.849	0.763	0.930	0.978	0.113	0.064	0.137	0.206	0.591	0.421	0.722	0.885
	StarE	0.314	0.233	0.350	0.469	-	-	-	-	0.150	0.084	0.169	0.307	-	-	-	-
	GRAN	0.444	0.380	0.477	0.562	0.948	0.920	0.974	0.990	0.209	0.153	0.230	0.314	0.718	0.609	0.803	0.893
DT-KGE	JOIE	0.181	0.108	0.218	0.315	-	-	-	-	0.124	0.077	0.140	0.215	-	-	-	-
	HyperKA	0.164	0.083	0.192	0.302	-	-	-	-	0.103	0.062	0.118	0.182	-	-	-	-
DH-KGE	DH w/o GE	0.044	0.022	0.051	0.081	0.193	0.091	0.193	0.414	0.015	0.005	0.013	0.030	0.709	0.595	0.793	0.856
	DH w/o HGA	0.450	0.382	0.485	0.573	0.955	0.930	0.978	0.990	0.219	0.175	0.237	0.301	0.696	0.624	0.751	0.807
	DH w/o JL	0.446	0.377	0.482	0.574	0.953	0.928	0.977	0.989	0.216	0.168	0.235	0.307	0.714	0.639	0.773	0.829
	DHGE	**0.453**	**0.388**	**0.486**	**0.575**	**0.958**	**0.932**	**0.982**	**0.994**	**0.227**	**0.180**	**0.242**	**0.320**	**0.745**	**0.670**	**0.807**	**0.916**

Table 4: Results of Link Prediction on JW44K-6K.

and design the cross-link loss function as follows:

$$\begin{aligned}\mathcal{L}_{\text{Cross}} =& \frac{1}{|\mathcal{H}_S|} \sum_{(h_S,t_S)\in\mathcal{H}_S\wedge\left(h_S,t'_S\right)\notin\mathcal{H}_S} \\ & [\gamma + \left\|f\left(\mathbf{U}^{\mathcal{G}_I}(h_S)\right) - \mathbf{U}^{\mathcal{G}_O}(t_S)\right\|_2 \\ & - \left\|f\left(\mathbf{U}^{\mathcal{G}_I}(h_S)\right) - \mathbf{U}^{\mathcal{G}_O}(t'_S)\right\|_2],\end{aligned} \tag{7}$$

where we use the 2-norm function to calculate the distance deviation of entities and concepts under the same embedding space, t'_S is the negative example of t_S, and γ is a parameter of margin.

5.3 Joint Learning on DH-KG

We combine the training losses of the instance view, ontology view, and cross-view link set to get a joint learning loss:

$$\mathcal{L}_{\text{overall}} = \mathcal{L}_{\text{Intra}}^{\mathcal{G}_I} + \mathcal{L}_{\text{Intra}}^{\mathcal{G}_O} + \omega \cdot \mathcal{L}_{\text{Cross}}\ , \tag{8}$$

and we adopt the Adam optimizer to optimize the three losses alternately, where ω distinguishes the learning rate of intra-view loss and cross-view loss to achieve joint learning on DH-KG.

6 Experiments

In this section, we evaluate our proposed model DHGE with the tasks of link prediction (LP), entity typing (ET), and hypertension disease medicine prediction (MP, MCP), respectively. We introduce our experimental setup, followed by experimental results.

6.1 Experimental Setup

Datasets Our experiments are done on two DH-KG datasets, JW44K-6K and HTDM mentioned above. On JW44K-6K, we evaluate the LP and ET tasks on DH-KGs. On HTDM, we only perform entity prediction of the object position in the instance view to simulate the prediction of the doctor's medication for real hypertensive patients.

Baselines We compare DHGE with the following representative KG embedding (KGE) methods. The first category is Single-view Triple-based KGE Methods (ST-KGE): **TransE** (Bordes et al. 2013), **RotatE** (Sun et al. 2019), **TuckER** (Balazevic, Allen, and Hospedales 2019), and **AttH** (Chami et al. 2020), which ignore the auxiliary attribute-value pairs of the hyper-relational facts and treat the dual view as a single view for the experiment. The second category is Single-view Hyper-relational KGE Methods (SH-KGE): **HINGE** (Rosso, Yang, and Cudré-Mauroux 2020), **StarE** (Galkin et al. 2020), and **GRAN** (Wang et al. 2021), which treats facts as a hyper-relational paradigm in a single view. The third category is Dual-view Triple-based KGE Methods (DT-KGE): **JOIE** (Hao et al. 2019), and **HyperKA** (Sun et al. 2020), which can learn dual-view information, but facts are also triple-based.

Ablations We ignore the three modules of DHGE: intra-layer GRAN encoder, HGNN neighbor information aggregation, and joint learning, respectively, to form three ablation models: **DHGE w/o GE**, **DHGE w/o HGA**, and **DHGE w/o JL** for ablation study.

Evaluation Metrics For the task of LP and ET, we choose mean reciprocal rank (MRR) and hits at 1, 3, and 10 (H@1, H@3, H@10) evaluation indicators introduced by Bordes et al. (2013). For Medicine Prediction (MP, MCP) on HTDM, we cancel the H@10 indicator for the requirement of high accuracy.

Hyperparameters and Enviroment Since hyperparameters are sensitive to experimental results, we use the following hyperparameters for all models: embedding dimension = 256 for both entity and relation, epoch = 100, learning rate = 5e−4, and batch size = 256. All experiments were conducted with a single 11GB GeForce GTX 1080Ti GPU, and all results were obtained by averaging 3 experiments.

Catagory	Model	JW44K-6K: Entity Typing			
		MRR	H@1	H@3	H@10
ST-KGE	TransE	0.085	0.031	0.102	0.204
	RotatE	0.362	0.293	0.390	0.495
	TuckER	0.405	0.350	0.429	0.501
	AttH	0.416	0.348	0.448	0.539
SH-KGE	HINGE	0.109	0.062	0.132	0.198
	StarE	0.380	0.321	0.404	0.479
	GRAN	0.422	0.343	0.460	0.571
DT-KGE	JOIE	0.603	0.535	0.645	0.773
	HyperKA	**0.692**	0.626	**0.736**	0.798
DH-KGE	DHGE w/o GE	0.492	0.380	0.560	0.695
	DHGE w/o HGA	0.662	0.627	0.675	0.702
	DHGE w/o JL	0.428	0.332	0.468	0.613
	DHGE	0.690	**0.637**	0.726	**0.805**

Table 5: Results of Entity Typing on JW44K-6K.

Catagory	Model	HTDM: MP			HTDM: MCP		
		MRR	H@1	H@3	MRR	H@1	H@3
ST-KGE	TransE	0.041	0.006	0.047	0.224	0.142	0.428
	RotatE	0.074	0.023	0.095	0.469	0.180	0.519
	TuckER	0.213	0.075	0.214	0.579	0.419	0.642
	AttH	0.170	0.071	0.190	0.526	0.357	0.576
SH-KGE	HINGE	0.189	0.084	0.189	0.624	0.571	0.642
	StarE	0.224	0.095	0.285	0.637	0.571	0.667
	GRAN	0.374	0.238	0.428	0.679	0.633	0.857
DT-KGE	JOIE	0.073	0.014	0.090	0.361	0.181	0.514
	HyperKA	0.062	0.013	0.072	0.323	0.119	0.461
DH-KGE	DH w/o GE	0.274	0.095	0.381	0.559	0.285	0.809
	DH w/o HGA	0.306	0.238	0.333	0.668	0.476	0.857
	DH w/o JL	0.301	0.190	0.333	0.628	0.428	0.761
	DHGE	**0.445**	**0.381**	**0.572**	**0.772**	**0.667**	**0.905**

Table 6: Results of Medicine Prediction on HTDM.

6.2 Main Results and Analysis

In this experiment, we compare the LP and ET results of DHGE with all baselines and ablations.

Link Prediction We have defined the LP task on DH-KG, which includes entity and relation prediction for instance and ontology views. The conclusions are drawn from comparing DHGE to baselines and ablations in Table 4.

Comparison with ST-KGE: DHGE outperforms the ST-KGE model in JW44K-6K LP by a large margin because triple-based embedding methods are inherently unable to model neither auxiliary attribute-value pairs of hyper-relational facts nor hierarchical information. **Comparison with SH-KGE:** The intra-view LP task results of DHGE are generally higher than that of SH-KGE because cross-view links influence the embedding in one view and guide the embedding in another to be more accurate. **Comparison with DT-KGE:** The intra-view LP task performance of JOIE and HyperKA are scanty, even worse than the ST-KGE models because they are triple-based KGE methods and focus more on the ET task across views. **Comparison with DH-KGE ablation model:** The performance of all ablations degrades, and DHGE w/o GE is the worst performer, indicating that GRAN Encoder, Hypergraph Neighbourhood Association, and Joint Learning are useful in LP, and GRAN Encoder plays the most critical role. **Comparison between instance view and ontology view results:** The LP task performance of the ontology view is much more deficient than that of the instance view, indicating that the LP of the ontology view is more complicated because it is difficult for humans to predict and understand the facts between concept entities.

Entity Typing In this paper, we define the ET task on DH-KG, that is, to correspond the concepts in the ontology view with the entities in the instance view. We compare DHGE with baselines and ablations in Table 5 and obtain the following conclusions.

Comparison with ST-KGE and SH-KGE: These two KGE methods use the single view for entity prediction tasks. In the ET task, DHGE with cross-view learning performs better than a single-view model. **Comparison with DT-KGE:** The performance of models of DT-KGE is close to that of DHGE because JW44K-6K has rich hierarchical information. DT-KGE models still have good performance though they are triple-based. **Comparison with DH-KGE ablation model:** The ablation model without joint training has the worst performance in the entity typing task, indicating that joint training is the most critical part of the ET task because of rich cross-view links on DH-KG.

Hypertension Medicine Prediction Here, we present an example of a medical application of DH-KGs: hypertension medication prediction. We define the 133 entities of medicine as a fine-grained medicine set and the 8 entities of medicine class as a coarse-grained medicine class set. Then, we define the prediction within these two sets as a medicine prediction task (MP) and a medicine class prediction task (MCP), respectively. These two experiments do the LP task on the hyper-relational tail entity position of the instance view through the joint learning of the DH-KG.

Table 6 shows that the prediction results of HTDM are consistent with the instance-view link prediction when comparing the DHGE model with the baseline and ablation model. The MP and MCP results of DHGE are significantly higher than baselines and ablations. DHGE's MCP Hits@1 can reach 0.667, and Hits@3 can reach 0.905, indicating the excellent application prospect of DHGE.

7 Conclusion

This paper focuses on dual-view hierarchical hyper-relational embedding. We introduce the tasks of link prediction and entity typing for DH-KGs and create two new datasets, JW44K-6K and HTDM. Our DHGE model captures hyper-relational and hierarchical information and outperforms related models. In addition, we also demonstrate a use case in hypertension medication.

Acknowledgements

This work is supported by the National Science Foundation of China (Grant No. 62176026, Grant No. 61902034) and Beijing Natural Science Foundation (M22009, L191012). This work is also supported by the BUPT Postgraduate Innovation and Entrepreneurship Project led by Haoran Luo.

References

Balazevic, I.; Allen, C.; and Hospedales, T. M. 2019. TuckER: Tensor Factorization for Knowledge Graph Completion. In Inui, K.; Jiang, J.; Ng, V.; and Wan, X., eds., *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, 5184–5193. Association for Computational Linguistics.

Bollacker, K. D.; Evans, C.; Paritosh, P. K.; Sturge, T.; and Taylor, J. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In Wang, J. T., ed., *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008*, 1247–1250. ACM.

Bordes, A.; Usunier, N.; García-Durán, A.; Weston, J.; and Yakhnenko, O. 2013. Translating Embeddings for Modeling Multi-relational Data. In Burges, C. J. C.; Bottou, L.; Ghahramani, Z.; and Weinberger, K. Q., eds., *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, 2787–2795.

Chami, I.; Wolf, A.; Juan, D.; Sala, F.; Ravi, S.; and Ré, C. 2020. Low-Dimensional Hyperbolic Knowledge Graph Embeddings. In Jurafsky, D.; Chai, J.; Schluter, N.; and Tetreault, J. R., eds., *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, 6901–6914. Association for Computational Linguistics.

Dettmers, T.; Minervini, P.; Stenetorp, P.; and Riedel, S. 2018. Convolutional 2D Knowledge Graph Embeddings. In McIlraith, S. A.; and Weinberger, K. Q., eds., *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, 1811–1818. AAAI Press.

Feng, Y.; You, H.; Zhang, Z.; Ji, R.; and Gao, Y. 2019. Hypergraph Neural Networks. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, 3558–3565. AAAI Press.

Galkin, M.; Trivedi, P.; Maheshwari, G.; Usbeck, R.; and Lehmann, J. 2020. Message Passing for Hyper-Relational Knowledge Graphs. In Webber, B.; Cohn, T.; He, Y.; and Liu, Y., eds., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, 7346–7359. Association for Computational Linguistics.

Guan, S.; Jin, X.; Guo, J.; Wang, Y.; and Cheng, X. 2021. Link Prediction on N-ary Relational Data Based on Relatedness Evaluation. *CoRR*, abs/2104.10424.

Hao, J.; Chen, M.; Yu, W.; Sun, Y.; and Wang, W. 2019. Universal Representation Learning of Knowledge Bases by Jointly Embedding Instances and Ontological Concepts. In Teredesai, A.; Kumar, V.; Li, Y.; Rosales, R.; Terzi, E.; and Karypis, G., eds., *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, 1709–1719. ACM.

Rosso, P.; Yang, D.; and Cudré-Mauroux, P. 2020. Beyond Triplets: Hyper-Relational Knowledge Graph Embedding for Link Prediction. In Huang, Y.; King, I.; Liu, T.; and van Steen, M., eds., *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, 1885–1896. ACM / IW3C2.

Sun, Z.; Chen, M.; Hu, W.; Wang, C.; Dai, J.; and Zhang, W. 2020. Knowledge Association with Hyperbolic Knowledge Graph Embeddings. In Webber, B.; Cohn, T.; He, Y.; and Liu, Y., eds., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, 5704–5716. Association for Computational Linguistics.

Sun, Z.; Deng, Z.; Nie, J.; and Tang, J. 2019. RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Vrandecic, D.; and Krötzsch, M. 2014. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10): 78–85.

Wang, Q.; Wang, H.; Lyu, Y.; and Zhu, Y. 2021. Link Prediction on N-ary Relational Facts: A Graph-based Approach. In Zong, C.; Xia, F.; Li, W.; and Navigli, R., eds., *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, 396–407. Association for Computational Linguistics.

Wen, J.; Li, J.; Mao, Y.; Chen, S.; and Zhang, R. 2016. On the Representation and Embedding of Knowledge Bases beyond Binary Relations. In Kambhampati, S., ed., *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, 1300–1307. IJCAI/AAAI Press.

Zhou, J.; Hu, S.; Lv, X.; Yang, C.; Liu, Z.; Xu, W.; Jiang, J.; Li, J.; and Sun, M. 2021. KACC: A Multi-task Benchmark for Knowledge Abstraction, Concretization and Completion. In Zong, C.; Xia, F.; Li, W.; and Navigli, R., eds., *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, 1751–1763. Association for Computational Linguistics.