Rule-Guided Compositional Representation Learning on Knowledge Graphs

Guanglin Niu,[1] **Yongfei Zhang,**[1,2] **Bo Li**[*],[1,2] **Peng Cui,**[3] **Si Liu,**[1] **Jingyang Li,**[1] **Xiaowei Zhang**[4]
[1]Beijing Key Laboratory of Digital Media, School of Computer Science and Engineering, Beihang University,
[2]State Key Laboratory of Virtual Reality Technology and Systems, Beihang University,
[3]Department of Computer Science and Technology, Tsinghua University,
[4]College of Computer Science and Technology, Qingdao University
beihangngl@buaa.edu.cn, yfzhang@buaa.edu.cn, boli@buaa.edu.cn, cuip@tsinghua.edu.cn
liusi@buaa.edu.cn, lijingyang@buaa.edu.cn, xiaowei19870119@sina.com

Abstract

Representation learning on a knowledge graph (KG) is to embed entities and relations of a KG into low-dimensional continuous vector spaces. Early KG embedding methods only pay attention to structured information encoded in triples, which would cause limited performance due to the structure sparseness of KGs. Some recent attempts consider paths information to expand the structure of KGs but lack explainability in the process of obtaining the path representations. In this paper, we propose a novel Rule and Path-based Joint Embedding (RPJE) scheme, which takes full advantage of the explainability and accuracy of logic rules, the generalization of KG embedding as well as the supplementary semantic structure of paths. Specifically, logic rules of different lengths (the number of relations in rule body) in the form of Horn clauses are first mined from the KG and elaborately encoded for representation learning. Then, the rules of length 2 are applied to compose paths accurately while the rules of length 1 are explicitly employed to create semantic associations among relations and constrain relation embeddings. Moreover, the confidence level of each rule is also considered in optimization to guarantee the availability of applying the rule to representation learning. Extensive experimental results illustrate that RPJE outperforms other state-of-the-art baselines on KG completion task, which also demonstrate the superiority of utilizing logic rules as well as paths for improving the accuracy and explainability of representation learning.

1 Introduction

Knowledge graphs (KGs) such as Freebase (Bollacker, Gottlob, and Flesca 2008), DBpedia (Lehmann et al. 2015) and NELL (Mitchell et al. 2018) are knowledge bases which store factual triples consisting of entities with their relations. They have achieved rapid development and extensive applications for various research fields, such as zero-shot recognition (Wang, Ye, and Gupta 2018), question answering (Hao et al. 2018) and recommender systems (Zhang et al. 2016).

Typical KGs contain billions of triples but remain to be incomplete. Specifically, 75% of 3 million person entities miss a nationality in Freebase (West et al. 2014) and 60% of person entities do not have a place of birth in DBpedia (Krompaß, Baier, and Tresp 2015). Thus, it is hard to further introduce KGs into some applications, such as no correct answers for question answering systems based on incomplete KGs. And the symbolic nature of triple facts in the form of (head entity, relation, tail entity) makes it challenging to expand large scale KGs.

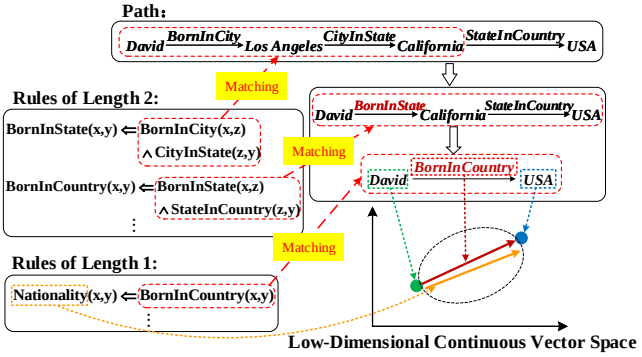

Figure 1: A motivating example of our proposed approach based on Horn rules and a path.

In recent years, representation learning on KGs (or known as KG embedding) such as TransE (Bordes et al. 2013), TransH (Wang et al. 2014) and TransR (Lin et al. 2015b) has become popular which intends to embed entities and relations of KGs into a continuous vector space while retaining the inherent structure and latent semantic information of KGs (Wang et al. 2017). This could benefit a lot for large scale KG completion.

The above embedding models purely consider the single triples. In fact, multi-step paths in KGs always play a pivotal role in providing extra relationships between entity pairs. For instance, we can utilize a path in the KG $David \xrightarrow{BornInState} California \xrightarrow{StateInCountry} USA$ to expand the knowledge for its corresponding triple fact $(David, Nationality, USA)$. Both Lin et al. (2015a) and Guu et al. (2015) succeeded to learn the entity and relation embeddings on paths. In their work, the relation embeddings are initialized randomly and the path representations are

*Corresponding author.

composed via addition, multiplication or Recurrent Neural Networks (RNN) of the relations along the paths. Since the path representations are achieved purely based on numerical calculations in a latent space, such approaches will cause error propagation and limited accuracy of the paths embeddings and further affect the whole representation learning.

An effective way to apply the extra semantic information in KG embedding is to employ the logic rules in view of their accuracy and explainability. KALE (Guo et al. 2016) and RUGE (Guo et al. 2018) both convert the rules into complex formulae modeled by t-norm fuzzy logics to transfer the knowledge in rules into the learned embeddings. Nevertheless, logic rules could maintain their original explainability only in the symbolic form and such rules actually focus on the semantic associations as well as the constraints of various relations, which have not been well exploited for the triples and paths in KG embedding.

This paper proposes a novel rule-guided compositional representation learning approach named Rule and Path-based Joint Embedding (RPJE), using Horn rules to compose paths and associate relations in the semantic level to improve the precision of learning KG embeddings on paths and enhance the explainability of our representation learning. In allusion to the path as shown in Figure 1, the rule bodies in two Horn rules of length 2 $borninstate(x, y) \Leftarrow bornincity(x, z) \wedge cityinstate(z, y)$ and $bornincounrty(x, y) \Leftarrow borninstate(x, z) \wedge stateincountry(z, y)$ are respectively matched with two segments of the path, which could be applied to iterately compose the entire path as a straightforward triple $(David, BornInCountry, USA)$. Then, the rule body of length 1 $Nationality(x, y) \Leftarrow BornInCountry(x, y)(0.9)$ is matched with the relation $BornInCountry$. Therefore, two relation embeddings denoting $Nationality$ and $BornInCountry$ are further constrained to be closer in the latent space with confidence level 0.9.

In experiments, we evaluate the proposed model RPJE on four benchmark datasets of FB15K, FB15K-237, WN18, and NELL-995. Experimental results illustrate that our approach achieves superior performances on KG completion task and significantly outperforms state-of-the-art baselines, which verifies the capability of combining rules with paths in KG embedding. Our main contributions of this work are:

- To the best of our knowledge, this is the first attempt to integrate logic rules with paths for KG embedding, endowing our model with both the explainability from semantic level and the generalization from data level.
- Our proposed model RPJE considers the various types of rules to inject prior knowledge into KG embedding. It can use the encoded length-2 rules to start paths composition rather than with randomly initialized vectors for obtaining accurate path representations. And the semantic associations among relations could be created by length-1 rules.
- We conduct extensive experiments on KG completion and our model achieves promising performances. The influence of different confidence thresholds of rules demonstrates that considering confidence of rules in our model guarantees the effectiveness of using rules and could achieve good robustness to various confidence thresholds.

2 Related Work

KG embedding models: In recent years, many works have been done to learn distributed representations for entities and relations in KGs, which fall into three major categories: (1) Translational distance model. Inspired by the translation invariant principle from word embedding (Mikolov, tau Yih, and Zweig 2013), TransE (Bordes et al. 2013) regards relations as translating operations between head and tail entities, i.e., the formula $h + r \approx t$ should be satisfied when triple (h, r, t) holds. (2) Tensor decomposition model. DistMult (Yang et al. 2015) and ComplEx (Trouillon et al. 2016) both utilize tensor decomposition to represent each relation as a matrix and each entity as a vector. 3) Neural networks model. In NTN (Socher et al. 2013), a 3-way tensor and two transfer matrices are encoded into multilayer neural network. Among these methods, TransE and plenty of its variants TransH (Wang et al. 2014), TransR (Lin et al. 2015b) and TransG (Xiao, Huang, and Zhu 2016) have become promising approaches for successfully capturing the semantics of KG symbols. However, the above methods merely consider the facts immediately observed in KGs and ignore extra prior knowledge to enhance KG embedding.

Path enhanced models: Paths existing in KGs have gained more attentions to be combined with KG embedding because multi-hop paths could provide relationships between seemingly unconnected entities in KGs. Path Ranking Algorithm (PRA) (Lao, Mitchell, and Cohen 2011) is one of the early studies which searches paths by random walk in KGs and regards the paths as features for a per-target relation binary classifier. Neelakantan et al. (2015) proposed a compositional vector space model with a recurrent neural network to model relational paths on knowledge graph completion. Guu et al. (2015) introduced additive and multiplicative interactions between relation matrices in the path. Lin et al. (2015a) proposed PTransE to obtain the path embeddings by composing all the relations in each path. DP-TransE (Zhang et al. 2018) jointly builds interactions between the latent features and graph features of KGs to offer precise and discriminative embedding. However, all these techniques obtain the path representations via calculating relation embeddings along the paths, which would cause limited accuracy and lack explainability.

Rule extraction and rule enhanced models: Logic rules are explainable and contain rich semantic information, which have shown the power in knowledge inference. The Inductive Logic Programming algorithms such as XAIL are available to learn FOL or even ASP-style rules. However, it is difficult to mine rules from KGs with ILP algorithms due to open world assumption of KGs (absent information cannot be taken as counterexamples). Therefore, several rule mining methods have been developed to extract rules efficiently from large scale KGs, including AMIE (Galárraga et al. 2013), AMIE+ (Galárraga et al. 2015), RLvLR (Omran, Wang, and Wang 2018) and CARL (Tanon et al. 2018).

Based on the valid rules, some studies integrate logical rules in deep neural networks (DNN) and show impressive results in sentence sentiment analysis and named entity

recognition (Hu et al. 2016a; Hu et al. 2016b). Markov logic network (MLN) (Richardson and Domingos 2006) combines first-order logic with probabilistic graphical models for reasoning but it is inefficient to infer via MLN approach and the performance is limited due to some triples cannot be discovered by any rules. Besides, some works (Gad-Elrab et al. 2016; Tran et al. 2016) obtain the answer set semantics by directly using horn rules on KGs for KG completion but lack high generalization and the main usage of these models is learning rules.

To improve both the precision and generalization of KG completion, some of the recent researches attempt to incorporate the rules into KG embedding models (Wang, Wang, and Guo 2015). Minervini et al. (2017) imposed the equivalence and inversion constraints on the relation embeddings. But this approach considers only two types of constraints between relations rather than general rules, which might not always be available for any KG. In both KALE (Guo et al. 2016) and RUGE (Guo et al. 2018), the triples are represented as atoms and the rules are modeled by t-norm fuzzy logic for being converted into complex formulae formed by atoms with logical connectives. However, the above two methods quantify the rules in embedding which would decrease the explainability and accuracy of rules. By eliminating the complex process of modeling rules to be formulae, we explicitly and immediately employ the Horn rules to deduce the path embeddings and create the semantic associations between relations.

3 Methodology

We attempt to integrate paths with logic rules to provide more semantic information in our model. The overall framework of the proposed scheme is shown in Figure 2. Firstly, we extract the paths and mine the Horn rules from KG, where the rules of length 1 and 2 are denoted as Rules R_1 and Rules R_2, respectively (§3.1). Then, we apply Rules R_2 to iteratively compose paths and Rules R_1 to create the semantic associations of some relation pairs (§3.2). Furthermore, vector initialization is used to transform the entities and relations in symbolic space into the vector space for training the KG embeddings. Finally, compositional representation learning is implemented for optimizing objective specific to triples, paths and associated relations pairs (§3.3, §3.4).

3.1 Logic Rules Extraction from KG

Horn rules could be mined automatically by any KG rule exaction algorithm or tool. In this paper, we first mine rules together with their confidence levels denoted as $\mu \in [0, 1]$ from KGs. And a rule with higher confidence level has higher possibility to hold. We limit the maximum length of rules to 2 for the efficiency of mining valid rules. Thus, rules are classified into two types according to their length: (1) **Rules R_1**. The set of length-1 rules is denoted as Rules R_1, which associating two relations in rule body and rule head. (2) **Rules R_2**. The set of rules with length 2 is denoted as R_2, which could be utilized to compose paths. Some examples of Rules R_1 and Rules R_2 are provided in Table 1.

Remark: The inverse version of each relation is always added in path-based approaches to constrain each path along one direction and improve the graph connectivity (Zhang et al. 2018). Therefore, given a triple (h, r, t), a reconstructed triple (t, r^{-1}, h) is defined to express the inverse relationship r^{-1} between entity t and entity h.

3.2 Rules Employment for Compositional Representation Learning

The Horn rules extracted from KGs could be utilized in two modules for compositional representation learning, including paths composition by Rules R_2 and relation pairs association by Rules R_1.

Paths Composition by Rules R_2. We first implement paths extraction procedure by PTransE (Lin et al. 2015a) on KGs, where each path p is extracted together with its reliability which is achieved by the path-constraint resource allocation mechanism and denoted as $R(p|h, t)$ between an entity pair (h, t). We generate each path set $P(h, t)$ by selecting the paths between the entity pair (h, t) with their reliability over 0.01. Specifically, it is essential to form a sequential path by atoms of each rule body in Rules R_2 for composing paths. A chain rule is further defined as the rule which the entity pair linked by the chain in the rule body is also connected by the relation in the rule head. However, the rules mined by most of the existing open-source rule mining systems could not be directly utilized because these rules are not chain rules. Therefore, we should encode each rule to form a directed path of its rule body (removing some of the rules could not be converted as this formalization in any case). In total, there are totally 8 different types of rules conversion modes, as provided in Table 2. Take the original rule $r_3(a, b) \Leftarrow r_1(\mathbf{e}, b) \wedge r_2(\mathbf{e}, a)$ for instance, we first convert the atom $r_2(\mathbf{e}, a)$ into $r_2^{-1}(a, \mathbf{e})$, and then exchange two atoms in the rule body to obtain a chain rule $r_3(a, b) \Leftarrow r_2^{-1}(a, \mathbf{e}) \wedge r_1(\mathbf{e}, b)$, which could be further abbreviated to $r_3 \Leftarrow (r_2^{-1}, r_1)$. Then, a path containing a sequence of relations $r_2^{-1} \rightarrow r_1$ could be composed as r_3.

To make the best of encoded rules, we should traverse the paths and conduct the composition operation iteratively in the semantic level until no relation could be composed by rules since two relations are composed every time and the composition result might be composed in the next step. Considering two types of scenarios in practical paths composition procedure: (1) The optimal scenario that all of the relations in a path could be iteratively composed by Rules R_2 and finally joined together as a single relation between entity pair. (2) The general scenario that some relations are unable to be composed based on Rules R_2, we will adopt the numerical operation such as addition for the embeddings of these relations. Besides, in allusion to the situation that more than single rule could be matched in the path simultaneously, such as two rules $U(a, b) \Leftarrow R(a, c) \wedge T(c, b)$ as well as $V(a, b) \Leftarrow R(a, c) \wedge T(c, b)$ are both activated, the rule with the highest confidence should be selected to compose the path. Specifically, we define the path composition result via the above procedure as $C(p)$ which is also denoted as the path embedding of the path p.

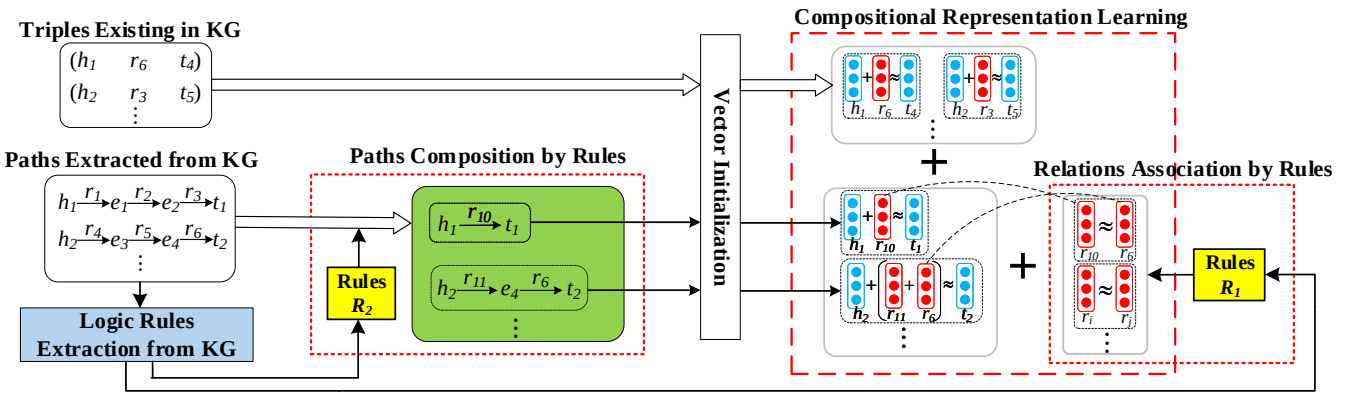

Figure 2: Overall architecture of our model.

Rules R_1 with confidence levels
$concept:citylocatedincountry(a,b) \Leftarrow concept:citycapitalofcountry(a,b)$ 0.86 (extracted from NELL-995)
$/location/location/people\_born\_here(a,b) \Leftarrow /people/person/place\_of\_birth^{-1}(a,b)$ 1 (extracted from FB15K)
Rules R_2 with confidence levels
$concept:parentofperson(a,b) \Leftarrow concept:hasspouse(a,e) \wedge concept:fatherofperson(e,b)$ 1 (extracted from NELL-995)
$/film/language(a,b) \Leftarrow /film/directed\_by(a,e) \wedge /person/language(e,b)$ 0.81 (extracted from FB15K)

Table 1: A few examples of rules mined from FB15K and NELL-995. The superscript "-1" means inverse relation.

The original rules	Encoded rules
$r_3(a,b) \Leftarrow r_1(a,\mathbf{e}) \wedge r_2(\mathbf{e},b)$	$r_3 \Leftarrow (r_1, r_2)$
$r_3(a,b) \Leftarrow r_1(\mathbf{e},b) \wedge r_2(a,\mathbf{e})$	$r_3 \Leftarrow (r_2, r_1)$
$r_3(a,b) \Leftarrow r_1(\mathbf{e},b) \wedge r_2(\mathbf{e},a)$	$r_3 \Leftarrow (r_2^{-1}, r_1)$
$r_3(a,b) \Leftarrow r_1(\mathbf{e},a) \wedge r_2(\mathbf{e},b)$	$r_3 \Leftarrow (r_1^{-1}, r_2)$
$r_3(a,b) \Leftarrow r_1(a,\mathbf{e}) \wedge r_2(b,\mathbf{e})$	$r_3 \Leftarrow (r_1, r_2^{-1})$
$r_3(a,b) \Leftarrow r_1(b,\mathbf{e}) \wedge r_2(a,\mathbf{e})$	$r_3 \Leftarrow (r_2, r_1^{-1})$
$r_3(a,b) \Leftarrow r_1(\mathbf{e},a) \wedge r_2(b,\mathbf{e})$	$r_3 \Leftarrow (r_1^{-1}, r_2^{-1})$
$r_3(a,b) \Leftarrow r_1(b,\mathbf{e}) \wedge r_2(\mathbf{e},a)$	$r_3 \Leftarrow (r_2^{-1}, r_1^{-1})$

Table 2: The list of conversion mode for Rules R_2. The left half are the original rules directly extracted from KG, and the right half are the encoded rules. In Table 2, variables a, b, e can be substituted by entities, where r_1, r_2 are denoted as relations in rule body and r_3 is the relation in rule head.

Relations Association by Rules R_1. On account of the Rules R_1, where a relation r_1 may have more semantic similarity with its directly implicating relation r_2 when the rule $\forall x, y : r_2(x,y) \Leftarrow r_1(x,y)$ holds. The rules in the form of $(a, r_2, b) \Leftarrow (b, r_1, a)$ need to be encoded as $(a, r_2, b) \Leftarrow (a, r_1^{-1}, b)$ for representation learning. Hence, in the training process, the embeddings denoting a pair of relations which appear simultaneously in Rules R_1 should be constrained to be closer than the embeddings of two relations that mismatch any rule.

3.3 Compositional Representation Modeling

Along with the strategy of translation-based algorithms, for each triple (h, r, t), we define three energy functions to respectively model correlations with the direct triple along with the typical translation-based methods, the path pair using Rules R_2 and the relation pair employing Rules R_1:

$$E_1(h,r,t) = \|\mathbf{h} + \mathbf{r} - \mathbf{t}\| \quad (1)$$

$$E_2(p,r) = R(p|h,t)\Big(\prod_{\mu_i \in B(p)} \mu_i\Big)\|C(p) - \mathbf{r}\| \quad (2)$$

$$E_3(r,r_e) = \|\mathbf{r} - \mathbf{r}_e\| \quad (3)$$

where $E_1(h,r,t)$ is defined with the less score if triple (h,r,t) holds. $E_2(p,r)$ denotes the energy function evaluating the similarity between path p and relation r, and $R(p|h,t)$ represents the reliability of the path p from the entity pair (h,t) and is calculated the same as in PTransE (Lin et al. 2015a). $\mathbf{h}$, $\mathbf{r}$ and $\mathbf{t}$ are the embeddings of head entity, relation and tail entity, respectively. $C(p)$ denotes the composition result of the path p, which is obtained according to the paths composition procedure explained in §3.2. And $B(p) = \{\mu_1, \ldots, \mu_n\}$ represents the set of confidence levels corresponding to all the rules in Rules R_2 employed in the process of composing the path p. $E_3(r,r_e)$ is the energy function indicating the similarity of relation r and another relation r_e and should be assigned with less score if r_e is the relation implicated by the relation r with Rules R_1. $\mathbf{r}_e$ is the embedding of the relation r_e.

3.4 Objective Formalization

With the open world assumption (Drumond, Rendle, and Schmidt-Thieme 2012), we introduce the pairwise ranking loss function to formalize our optimization objective of

RPJE for training, which is defined as

$$L = \sum_{(h,r,t)\in T} [L_1(h,r,t) + \alpha_1 \sum_{p\in P(h,t)} L_2(p,r) + \alpha_2 \sum_{r_e\in D(r)} L_3(r,r_e)] \quad (4)$$

In Eq.4, $D(r)$ is defined as the set of relations all deduced from r on the basis of Rules R_1, and r_e is any relation in $D(r)$. $P(h,t)$ denotes all the paths linking entity pair (h,t), and p is one of the path in $P(h,t)$. $L_1(h,r,t)$, $L_2(p,r)$ and $L_3(r,r_e)$ are three margin-based loss functions considering the energy functions in Eqs.1,2,3 to measure the effectiveness of representation learning in regard to the direct triple (h,r,t), the path pair (p,r) as well as the relation pair (r,r_e), respectively, which are defined as follows:

$$L_1(h,r,t) = \sum_{(h',r',t')\in T^-} max(0, \gamma_1 + E_1(h,r,t) - E_1(h',r',t')) \quad (5)$$

$$L_2(p,r) = \sum_{(r')\in T^-} max(0, \gamma_2 + E_2(p,r) - E_2(p,r') \quad (6)$$

$$L_3(r,r_e) = \sum_{(r')\in T^-} max(0, \gamma_3 + \beta E_3(r,r_e) - E_3(r,r')) \quad (7)$$

where the function $max(0,x)$ is defined to obtain the maximum value between 0 and x. γ_1, γ_2, γ_3 are three positive hyper-parameters denoting each margin of the loss functions in Eqs.5,6,7, respectively. The weight of triples is fixed to 1, and α_1, α_2 are two hyper-parameters respectively weighting the influence of paths and relation pairs embedding constraint. β denotes the confidence level of the rule in Rules R_1 associating r and r_e. The confidence levels of all the rules are considered to be penalty coefficients in optimization. T represents a set that contains all the positive triples observed in KG. Following the negative sampling method as in (Bordes, Weston, and Bengio 2014), T^- contains the negative triples reconstructed via randomly replacing the entities and relations in T and removing the triples already exist in T.

$$T^- = (h',r,t) \cup (h,r',t) \cup (h,r,t') \quad (8)$$

To solve the optimization, we utilize mini-batch stochastic gradient descent (SGD). And considering the training efficiency, the paths are limited no longer than 3 steps.

4 Experiments

4.1 Experiment Settings

Datasets and Rules. We evaluate our model on four typical datasets: FB15K and FB15K-237 both extracted from the large-scale Freebase (Bollacker, Gottlob, and Flesca 2008), WN18 extracted from WordNet (Miller 1995) and NELL-995 extracted from NELL (Mitchell et al. 2018). Note that FB15K-237 contains no inverse relation and hence it is hard to learn embeddings by these mutually independent relations, so FB15K and FB15K-237 are always regarded as two distinguishing datasets. Statistics of datasets used are shown in Table 3. We evaluate the performance of our approach and other baselines on KG completion task, which is specifically formulated as entity prediction and relation prediction. Specifically, entity prediction aims to complete a triple with one entity missing while relation prediction aims to predict a relation given head and tail entities.

Dataset	#Rel	#Ent	#Train	#Valid	#Test
FB15K	1,345	14,951	483,142	50,000	59,071
FB15K-237	237	14,541	272,115	17,535	20,466
WN18	18	40,943	141,442	5,000	5,000
NELL-995	200	75,492	123,370	15,000	15,838

Table 3: Statistics of datasets used in the experiments. Rel denotes relation and Ent denotes entity.

Datasets	Rule Types	Various Confidence Thresholds #0.5	#0.6	#0.7	#0.8	#0.9
FB15K	R_1	1,157	975	899	767	586
	R_2	643	632	586	535	229
FB15K-237	R_1	93	89	81	76	40
	R_2	359	309	292	253	232
WN18	R_1	17	17	17	17	16
	R_2	89	80	77	24	0
NELL-995	R_1	132	96	58	40	15
	R_2	326	266	201	161	105

Table 4: Statistics of the encoded rules in various confidence thresholds from the four datasets. Note that the rules in Rules R_1 extracted from WN18 have the nearly same amount for their confidence levels all exceed 0.8.

Our scheme is readily incorporable to any rule mining tool. And we choose AMIE+ (Galárraga et al. 2015) for its convenience and fast-speed to mine rich rules with an alternative confidence threshold on different databases. The confidence thresholds of rules are selected in the range of [0,1] with the step size 0.1 to search the best performance of rules on datasets. Table 4 lists the statistics of rules with various confidence thresholds in the range of $[0.5, 0.9]$ mined from FB15K, FB15K-237, WN18 as well as NELL-995, which have been encoded for representation learning.

Evaluation Protocols. Three principle assessment metrics are focused on: the mean rank of correct entities (MR), the mean reciprocal rank of correct entities (MRR) and the proportion of test triples for which correct entity is ranked in the top n predictions (Hits@n). And an evaluation result should achieve lower MR, higher MRR and Hits@10. Moreover, the "filtered" setting eliminates the reconstructed triples that could be observed in the KG, yet the "raw" setting does not. To achieve these metrics, We define the score function for calculating the scores for reconstructed triples as follows:

$$Q(h,r,t) = \|\mathbf{h} + \mathbf{r} - \mathbf{t}\| + \alpha_1 \sum_{p\in P(h,t)} R(p|h,t)(\prod_{\mu_i\in B(p)} \mu_i)\|C(p) - \mathbf{r}\| \quad (9)$$

As shown in Eq.9, the Rules R_2 should be utilized for composing paths in testing process. We rank the scores in descending order.

Baselines for Comparison. To verify the performance of our approach, we select several involved state-of-the-art models to implement KG completion, including three types of baselines: (1) Embedding methods only considering triple facts: TransE (Bordes et al. 2013), TransH (Wang et al. 2014), TransR (Lin et al. 2015b), STransE (Nguyen et al. 2016), TransG (Xiao, Huang, and Zhu 2016), TEKE (Wang and Li 2016), R-GCN+ (Michael et al. 2018), KB-LRN (Garciaduran and Niepert 2017), ConvE (Dettmers et al. 2018). (2) Path-based models: PTransE (Lin et al. 2015a) and DPTransE (Zhang et al. 2018). (3) Rule enhanced models: KALE (Guo et al. 2016) and RUGE (Guo et al. 2018). We use the best results presented in their original papers and also implement PTransE and RUGE by their source codes.

Experimental Settings. To guarantee fair comparison, we adopt the following evaluation settings in our work: (1) 100 mini-batches are created on datasets. (2) The entity and relation embeddings are initialized randomly and limited to unit vectors. (3) Following the same configurations as many prevailing baselines, the learning rate is chosen as 0.001, γ_1 and γ_2 are selected as $\gamma_1 = \gamma_2 = 1$, the embedding dimension is set to 50 for WN18 and 100 for other three datasets considering only 18 relations exist in WN18, dissimilarity is selected as L_1 and training epochs is set to 500. In addition, we employ a grid search to select the other optimal hyper-parameters. We manually tune the margin γ_3 in $\{1, 1, 5, 2, 2.5, 3\}$, and the weight coefficients α_1, α_2 both in $\{0, 5, 1, 1.5, 2, 3, 5\}$. The best models are selected on validation sets. The resulting optimal of margin γ_3 and the weight coefficients α_1, α_2 are assigned to: $\gamma_3 = 1, \alpha_1 = 1, \alpha_2 = 3$.

4.2 Influence of Confidence Levels and Path Steps

In this subsection, we experimentally examine the impact of the two important parameters in our proposed scheme, namely the confidence levels of the rules and path steps. It indicates that our model is robust to the noisy rules for the rules with low confidence will be filtered out according to the appropriate confidence threshold. For instance, based on the confidence threshold 0.7, the rule $AthletePlaysInLeague(x, y) \Leftarrow AthleteledTeam(x, z) \wedge TeamPlaysInLeague(z, y)$ with confidence 0.96 will be used in path composition, but the rule $PersonHasJobPosition(x, y) \Leftarrow HasSibling(x, z) \wedge PersonHasJobPosition(z, y)$ with confidence 0.6 will be removed. It is also known that selecting a confidence threshold is a trade-off between higher confidence with more rules. We investigate the performance influence by varying the confidence thresholds in the range of [0.1, 1.0] with step 0.1. From Figure 3, we can observe that the confidence thresholds of 0.7 and 0.8 achieve the best tradeoffs. Furthermore, RPJE outperforms PTransE with the confidence threshold in a broad range of [0.4, 1.0] which illustrates the rules exploited in our model will be effective as long as the confidence threshold is selected in a moderate range. Particularly, RPJE-min obtains worse performance with lower confidence threshold due to more incorrect rules employed will cut down the accuracy of representation learning.

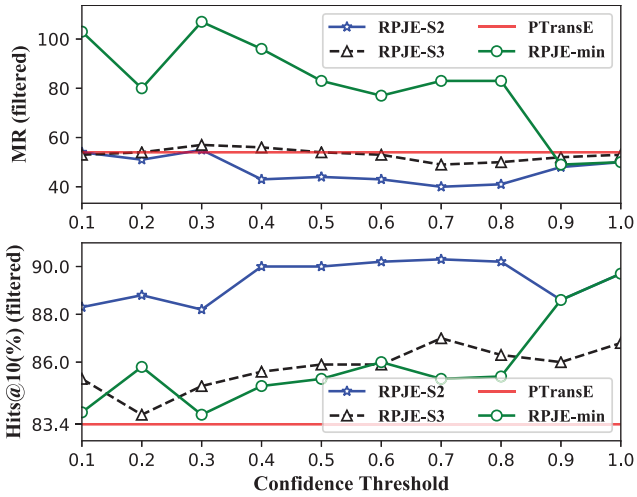

Figure 3: Performance comparison of various confidence thresholds and path steps limitation.

Models	MR raw	MR filtered	MRR filtered	Hits@ 10(%) raw	Hits@ 10(%) filtered
TransE	243	125	0.4	34.9	47.1
TransH	212	87	-	45.7	64.4
TransR	198	77	-	48.2	68.7
STransE	219	69	0.543	51.6	79.7
R-GCN+	-	-	0.696	-	84.2
KB-LRN	-	44	0.794	-	87.5
ConvE	-	51	0.657	-	83.1
PTransE	200	54	0.679	51.8	83.4
DPTransE	191	51	-	**58.1**	88.5
KALE	-	-	0.523	-	76.2
RUGE	191	71	0.768	54.3	86.5
RPJE	**186**	**40**	**0.816**	55.0	**90.3**

Table 5: Entity prediction results on FB15K. The missing values indicate the scores not reported in the original work. The best score is in **bold**, and the second best in underline.

Additionally, we could compare the performance of different path steps limitation between steps 2 and 3. Figure 3 illustrates the entity prediction results considering different confidence thresholds achieved by RPJE-S2 (2-step path), RPJE-S3 (3-step path), RPJE-min (RPJE ignoring the confidence of rules) and PTransE on FB15K. These results verify the confidence levels' contribution to representation learning. On account of same configurations, RPJE employing paths with maximum 2 steps consistently outperforms that with maximum 3 steps. The reason might be that longer paths may cause lower accuracy in paths composition, which will be studied in the future work. Therefore, we select the confidence threshold as 0.7 and the path steps as 2 for the best setting in the following results.

Models	Head Prediction (Hits@10)				Tail Prediction (Hits@10)			
	1-1	1-N	N-1	N-N	1-1	1-N	N-1	N-N
TransE	35.6	62.6	17.2	37.5	34.9	14.6	68.3	41.3
TransH	66.8	87.6	28.7	64.5	65.5	39.8	83.3	67.2
TransR	78.8	89.2	34.1	69.2	79.2	37.4	90.4	72.1
STransE	82.8	94.2	50.4	80.1	82.4	56.9	93.4	83.1
TEKE	78.8	89.3	54.0	81.7	79.2	59.2	90.4	83.5
TransG	93.0	96.0	62.5	86.8	92.8	68.1	94.5	88.8
PTransE	91.0	92.8	60.9	83.8	91.2	74.0	88.9	86.4
DPTransE	92.5	95.0	58.0	86.6	93.5	71.1	93.9	88.2
RPJE	**94.2**	**96.5**	**70.4**	**91.6**	**94.1**	**83.9**	**95.3**	**93.3**

Table 6: Entity prediction results on FB15K by mapping properties of relations (%) with filtered setting.

Models	MR		Hits@1(%)	
	raw	filtered	raw	filtered
TransE	2.79	2.43	68.3	87.2
PTransE	1.81	1.35	69.5	93.6
RUGE	2.47	2.22	68.8	87.0
RPJE	**1.68**	**1.24**	**70.1**	**95.3**

Table 7: Relation prediction results on FB15K. We use Hits@1 for better comparison because Hits@10 of all the models exceed 95%.

Models	MR	MRR	Hits@10(%)
TransE	347	0.294	46.4
R-GCN+	-	0.249	41.7
KB-LRN	209	0.309	49.3
ConvE	246	0.316	41.7
PTransE	302	0.363	52.6
RUGE	488	0.164	34.9
RPJE	**207**	**0.470**	**62.5**

Table 8: Entity prediction results on FB15K-237.

Models	WN18		NELL-995	
	MRR	Hits@10(%)	MRR	Hits@10(%)
TransE	0.495	93.4	0.219	35.2
PTransE	0.890	94.5	0.304	43.7
RUGE	0.943	94.4	0.318	43.3
RPJE	**0.946**	**95.1**	**0.361**	**50.1**

Table 9: Entity prediction results on WN18 and NELL-995.

4.3 Evaluation Results on FB15K and FB15K-237

In this section, we first evaluate entity prediction and relation prediction of the proposed RPJE with a variety of baselines on FB15K. From Table 5, it can be observed that: (1) Our approach RPJE achieves superiority compared with other baselines, and most of the improvements are statistically significant. This demonstrates that RPJE learns more reasonable embeddings for KGs via using logic rules in conjunction with paths. (2) In particular, RPJE outperforms PTransE on each metric, which indicates the superiority of introducing logic rules for providing higher accuracy in paths composition and learning better path embeddings. (3) Compared to the rule-based baselines KALE and RUGE, RPJE obtains the improvements of 56.0%/6.3% on MRR and 18.5%/4.4% on Hits@10 (filtered), which demonstrates the effectiveness of explicitly employing rules for preserving more semantic information and further integrating paths.

Table 6 shows the evaluation results of predicting entities by various types of relations. We can observe that: 1) RPJE outperforms all baselines significantly and consistently in regard to all the relation categories. Compared to the best performing baseline TransG, RPJE achieves an average improvement of 4.2% in head entities prediction and 6.5% in tail entities prediction. 2) More interestingly, on the two toughest tasks of predicting head entities of N-1 relation and predicting tail entities of 1-N relation, our approach achieves the best performance improvements approximately 12.6% and 13.4% compared to the best baselines, respectively.

The results of relation prediction are shown in Table 7. Three typical models representing three types of baselines are implemented. The results illustrate that RPJE outperforms baselines in all metrics. It verifies that paths could provide extra relationships for entity pairs and rules can further create more semantic association for relations to improve relation embeddings and benefit for relation prediction.

Furthermore, we implement the experiments on dataset FB15K-237. Since FB15K-237 is constructed up to date, only a minority of existing works have implemented their experiments and show evaluation results on this dataset, which can be selected as baselines. As shown in Table 8, RPJE obtains the best performance with approximately 29.5% improvement compared to PTransE on MRR and 26.8% improvement compared to KB-LRN on Hits@10. Although no inverse relation could be observed in FB15K-237, we could employ Horn rules to provide significant supplements for building semantic associations of relations.

4.4 Evaluation Results on WN18 and NELL-995

We also test the models on datasets WN18 and NELL-995. Three types of typical models are selected as baselines. Few rules can be mined from WN18 due to extremely limited amount of relations. And very parse paths can be extracted

Models	MR		MRR	Hits@ 10(%)	
	raw	filtered	filtered	raw	filtered
RPJE	**186**	**40**	**0.816**	**55.0**	**90.3**
-PaRu2	205	63	0.453	49.3	72.1
-Ru1	193	47	0.812	54.5	90.0

Table 10: Ablation study by removing paths and length-2 rules as well as length-1 rules.

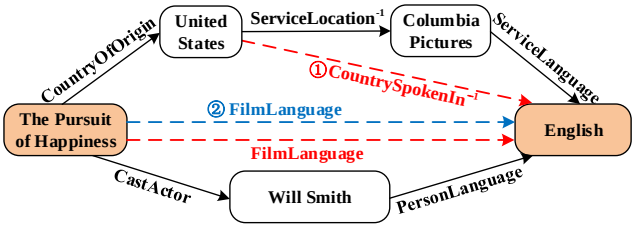

Figure 4: An example of the explainable relation prediction.

from NELL-995 because entities are far more than relations on this dataset. Even so, our model RPJE achieves consistent and significant improvements over other baselines as shown in Table 9. It illustrates the superiority of our approach for representation learning on various large scale KGs. On the other hand, considering the performance gains on FB15K are more than that on WN18, which is because more rules provide more semantic information to RPJE to use. As can be expected, our model RPJE will obtain better performance on datasets which implies more rules and paths.

4.5 Ablation Study

To verify the effectiveness of different components of RPJE, we implement the ablation study of entity prediction on FB15K by removing the paths as well as the length-2 rules at one time (-PaRu2), and the rules of length 1 (-Ru1) from our integrated model, respectively. More specifically, -PaRu2 means removing E_2/L_2 in Eq. 2/6 and -Ru1 means removing E_3/L_3 in Eq. 3/7. As shown in Table 10, we can conclude removing each component will lead to performance degradation especially when removing the paths and the rules of length 2 changes the results significantly.

4.6 Case Study

As shown in Figure 4, considering a relation prediction task with the given head entity $ThePursuitofHappiness$ and tail entity $English$, the result $FilmLanguage$ is obtained by our model RPJE. Particularly, this result can be explained by RPJE with paths and rules: for the 2-steps path, the rule

$$filmlanguage(x, y) \Leftarrow castactor(x, z) \wedge personlanguage(z, y)$$

with the confidence 0.81 is activated to compose the path into the prediction result $FilmLanguage$ while providing the confidence level 0.81 of this result. For the path of 3 steps, the intermediate composition result $CountrySpokenIn^{-1}$ is obtained by calling Rules R_2 and further employed to achieve the relation prediction result $FilmLanguage$ via embeddings on the reconstructed path containing the relation $CountrySpokenIn^{-1}$.

5 Conclusion and Future Work

In this paper, we proposed a novel model RPJE to learn KG embeddings by integrating triple facts, Horn rules and paths in a unified framework to enhance the accuracy and the explainability of representation learning. The Experimental results on KG completion verified the rules with confidence levels are significant in improving the accuracy of composing paths and enhancing the association between relations.

For future work, we will investigate some other potential composition operations such as Long Short Term Memory networks (LSTM) with attention mechanism which may benefit for long paths. And we will explore to push the embedding information back from RPJE to rule learning with a well-designed closed-loop system.

6 Acknowledgements

This work was partially supported by the National Natural Science Foundation of China (No. 61772054), the NSFC Key Project (No. 61632001), the Natural Science Foundation of China (No. 61572493) and the Fundamental Research Funds for the Central Universities.

References

[Bollacker, Gottlob, and Flesca 2008] Bollacker, K.; Gottlob, G.; and Flesca, S. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *KDD 2008*, 1247–1250.

[Bordes et al. 2013] Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; and Yakhnenko, O. 2013. Translating embeddings for modeling multi-relational data. In *NIPS 2013*, 2787–2795.

[Bordes, Weston, and Bengio 2014] Bordes, A.; Weston, G.; and Bengio, Y. 2014. A semantic matching energy function for learning with multi-relational data. *Machine Learning* 94:233–259.

[Dettmers et al. 2018] Dettmers, T.; Minervini, P.; Stenetorp, P.; and Riedel, S. 2018. Convolutional 2D knowledge graph embeddings. In *AAAI 2018*, 1811–1818.

[Drumond, Rendle, and Schmidt-Thieme 2012] Drumond, L.; Rendle, S.; and Schmidt-Thieme, L. 2012. Predicting rdf triples in incomplete knowledge bases with tensor factorization. In *SAC 2012*, 326–331.

[Gad-Elrab et al. 2016] Gad-Elrab, M. H.; Stepanova, D.; Urbani, J.; and Weikum, G. 2016. Exception-enriched rule learning from knowledge graphs. In *ISWC 2016*, 234–251.

[Galárraga et al. 2013] Galárraga, L.; Teflioudia, C.; Hose, K.; and Suchanek, F. 2013. AMIE: Association rule mining under incomplete evidence in ontological knowledge bases. In *WWW 2013*, 413–422.

[Galárraga et al. 2015] Galárraga, L.; Teflioudi, C.; Hose, K.; and Suchanek, F. 2015. Fast rule mining in ontological knowledge bases with AMIE+. *The VLDB Journal* 24(6):707–730.

[Garciaduran and Niepert 2017] Garciaduran, A., and Niepert, M. 2017. KBLRN: End-to-end learning of knowledge base representations with latent, relational, and numerical features. *arXiv* abs/1709.04676.

[Guo et al. 2016] Guo, S.; Wang, Q.; Wang, L.; Wang, B.; and Guo, L. 2016. Jointly embedding knowledge graphs and logical rules. In *EMNLP 2016*, 192–202.

[Guo et al. 2018] Guo, S.; Wang, Q.; Wang, L.; Wang, B.; and Guo, L. 2018. Knowledge graph embedding with iterative guidance from soft rules. In *AAAI 2018*, 4816–4823.

[Guu, Miller, and Liang 2015] Guu, K.; Miller, J.; and Liang, P. 2015. Traversing knowledge graphs in vector space. In *EMNLP 2015*, 318–327.

[Hao et al. 2018] Hao, Y.; Liu, H.; He, S.; Liu, K.; and Zhao, J. 2018. Pattern-revising enhanced simple question answering over knowledge bases. In *COLING 2018*, 3272–3282.

[Hu et al. 2016a] Hu, Z.; Ma, X.; Liu, Z.; Hovy, E.; and Xing, E. 2016a. Harnessing deep neural networks with logic rules. In *ACL 2016*, 2410–2420.

[Hu et al. 2016b] Hu, Z.; Yang, Z.; Salakhutdinov, R.; and Xing, E. P. 2016b. Deep neural networks with massive learned knowledge. In *EMNLP 2016*, 1670–1679.

[Krompaß, Baier, and Tresp 2015] Krompaß, D.; Baier, S.; and Tresp, V. 2015. Type-constrained representation learning in knowledge graphs. In *ISWC 2015*, 640–655.

[Lao, Mitchell, and Cohen 2011] Lao, N.; Mitchell, T.; and Cohen, W. W. 2011. Random walk inference and learning in a large scale knowledge base. In *EMNLP 2011*, 529–539.

[Lehmann et al. 2015] Lehmann, J.; Isele, R.; Jakob, M.; Anja Jentzsch, Dimitris Kontokostas, P. N. M. S. H. M. M. P. v. K. S. A.; and Bizer, C. 2015. Dbpedia-a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web* 6(2):167–195.

[Lin et al. 2015a] Lin, Y.; Liu, Z.; Luan, H.; Sun, M.; Rao, S.; and Liu, S. 2015a. Modeling relation paths for representation learning of knowledge bases. In *EMNLP 2015*, 705–714.

[Lin et al. 2015b] Lin, Y.; Liu, Z.; Sun, M.; Liu, Y.; and Zhu, X. 2015b. Learning entity and relation embeddings for knowledge graph completion. In *AAAI 2015*, 2181–2187.

[Michael et al. 2018] Michael, S.; N., K. T.; Peter, B.; van den Berg Rianne; Ivan, T.; and Max, W. 2018. Modeling relational data with graph convolutional networks. In *ESWC 2018*.

[Mikolov, tau Yih, and Zweig 2013] Mikolov, T.; tau Yih, W.; and Zweig, G. 2013. Linguistic regularities in continuous space word representations. In *NAACL-HLT 2013*, 746–751.

[Miller 1995] Miller, G. A. 1995. Wordnet: A lexical database for english. *Communications of the ACM* 38(11):39–41.

[Minervini et al. 2017] Minervini, P.; Costabello, L.; Munoz, E.; Novacek, V.; and Vandenbussche, P.-Y. 2017. Regularizing knowledge graph embeddings via equivalence and inversion axioms. In *ECMLPKDD 2017*, 668–683.

[Mitchell et al. 2018] Mitchell, T.; Cohen, W.; Hruschka, E.; Talukdar, P.; Yang, B.; Betteridge, J.; Carlson, A.; Dalvi, B.; Gardner, M.; Kisiel, B.; Krishnamurthy, J.; Lao, N.; Mazaitis, K.; Mohamed, T.; Nakashole, N.; Platanios, E.; Ritter, A.; Samadi, M.; Settles, B.; Wang, R.; Wijaya, D.; Gupta, A.; Chen, X.; Saparov, A.; Greaves, M.; and Welling, J. 2018. Never-ending learning. *Communications of the ACM* 61(5):103–115.

[Neelakantan, Roth, and McCallum 2015] Neelakantan, A.; Roth, B.; and McCallum, A. 2015. Compositional vector space models for knowledge base completion. In *ACL 2015*, 156–166.

[Nguyen et al. 2016] Nguyen, D. Q.; KairitSirts; Qu, L.; and Johnson, M. 2016. Stranse: a novel embedding model of entities and relationships in knowledge bases. In *NAACL-HLT 2016*, 460–466.

[Omran, Wang, and Wang 2018] Omran, P. G.; Wang, K.; and Wang, Z. 2018. Scalable rule learning via learning representation. In *IJCAI-ECAI-18*, 2149–2155.

[Richardson and Domingos 2006] Richardson, M., and Domingos, P. 2006. Markov logic networks. *Machine learning* 62(1-2):107–136.

[Socher et al. 2013] Socher, R.; Chen, D.; Manning, C. D.; and Ng, A. Y. 2013. Reasoning with neural tensor networks for knowledge base completion. In *NIPS 2013*, 926–934.

[Tanon et al. 2018] Tanon, T. P.; Stepanova, D.; Razniewski, S.; Mirza, P.; and Weikum, G. 2018. Completeness-aware rule learning from knowledge graphs. In *IJCAI 2018*, 5339–5343.

[Tran et al. 2016] Tran, H. D.; Stepanova, D.; Gad-Elrab, M. H.; Lisi, F. A.; and Weikum, G. 2016. Towards nonmonotonic relational learning from knowledge graphs. In *ILP 2016*, 94–107.

[Trouillon et al. 2016] Trouillon, T.; Welbl, J.; Riedel, S.; Éric Gaussier; and Bouchard, G. 2016. Complex embeddings for simple link prediction. In *ICML 2016*, 2071–2080.

[Wang and Li 2016] Wang, Z., and Li, J. 2016. Text-enhanced representation learning for knowledge graph. In *IJCAI 2016*, 1293–1299.

[Wang et al. 2014] Wang, Z.; Zhang, J.; Feng, J.; and Chen, Z. 2014. Knowledge graph embedding by translating on hyperplanes. In *AAAI 2014*, 1112–1119.

[Wang et al. 2017] Wang, Q.; Mao, Z.; Wang, B.; and Guo, L. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Dada Engineering* 29(12):2724–2743.

[Wang, Wang, and Guo 2015] Wang, Q.; Wang, B.; and Guo, L. 2015. Knowledge base completion using embeddings and rules. In *IJCAI 2015*, 1859–1865.

[Wang, Ye, and Gupta 2018] Wang, X.; Ye, Y.; and Gupta, A. 2018. Zero-shot recognition via semantic embeddings and knowledge graphs. In *CVPR 2018*, 6857–6866.

[West et al. 2014] West, R.; Gabrilovich, E.; Murphy, K.; Sun, S.; Gupta, R.; and Lin, D. 2014. Knowledge base completion via search based question answering. In *WWW 2014*, 515–526.

[Xiao, Huang, and Zhu 2016] Xiao, H.; Huang, M.; and Zhu, X. 2016. Transg: a generative model for knowledge graph embedding. In *ACL 2016*, 2316–2325.

[Yang et al. 2015] Yang, B.; tau Yih, W.; He, X.; Gao, J.; and Deng, L. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *ICLR 2015*.

[Zhang et al. 2016] Zhang, F.; Yuan, N. J.; Lian, D.; Xie, X.; and Ma, W.-Y. 2016. Collaborative knowledge base embedding for recommender systems. In *KDD 2016*, 353–362.

[Zhang et al. 2018] Zhang, M.; Wang, Q.; Xu, W.; Li, W.; ; and Sun, S. 2018. Discriminative path-based knowledge graph embedding for precise link prediction. In *ECIR 2018*, 276–288.