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 Zhang⁴
¹Beijing Key Laboratory of Digital Media, School of Computer Science and Engineering, Beihang University,
²State Key Laboratory of Virtual Reality Technology and Systems, Beihang University,
³Department of Computer Science and Technology, Tsinghua University,
⁴College 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.

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 BornInCountry California BornInState 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
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 \( \text{bornincity}(x,y) \Leftarrow \text{bornincity}(x,z) \land \text{cityincountry}(z,y) \) and \( \text{bornincountry}(x,y) \Leftarrow \text{bornincity}(x,z) \land \text{stateincountry}(z,y) \) are respectively matched with two segments of the path, which could be applied to iterately compose the entire path as a straightforward triple \((\text{David}, \text{BornInCountry}, \text{USA})\). Then, the rule body of length 1 \( \text{Nationality}(x,y) \Leftarrow \text{BornInCountry}(x,y) \) is matched with the relation \( \text{BornInCountry} \). Therefore, two relation embeddings denoting \( \text{Nationality} \) and \( \text{BornInCountry} \) 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, 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. DPTransE (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...
rules are classified into two types according to their length: (1) rules to 2 for the efficiency of mining valid rules. Thus, rules from KGs. And a rule with higher confidence level has to be iteratively composed paths and relations in symbolic space into the vector space for training the KG embeddings. Finally, compositional representation learning is implemented for optimizing objective functions of rules conversion modes, as provided in Table 2. Take the rule with the highest confidence should be selected to compose 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 functions specific to triples, paths and associated relations (§3.3, §3.4).

3.1 Logic Rules Extraction from KG

Horn rules could be mined automatically by any KG rule extraction algorithm or tool. In this paper, we first mine rules together with their confidence levels denoted as $\mu \in [0, 1]$ from KGs. 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 $r_3(a, b) \leftarrow r_1(e, b) \land r_2(e, a)$ for instance, we first convert the atom $r_2(e, a)$ into $r_2^{-1}(a, e)$, and then exchange two atoms in the rule body to obtain a chain rule $r_3(a, b) \leftarrow r_2^{-1}(a, e) \land r_1(e, b)$, 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 $U(a, b) \leftarrow R(a, c) \land T(c, b)$ as well as $V(a, b) \leftarrow R(a, c) \land T(c, b)$ 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.
Paths Extracted from KG

Paths Composition by Rules

Rules Composition by Rules

Triples Existing in KG

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 \(R_1\):

\[
E_1(h, r, t) = \|h + r - t\| \tag{1}
\]

\[
E_2(p, r) = R(p|h, t)(\prod_{\mu_i \in B(p)} \mu_i)|C(p) - r| \tag{2}
\]

\[
E_3(r, r_e) = \|r - r_e\| \tag{3}
\]

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). \(h, r\) and \(t\) 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 \(R_1\). \(r_e\) 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
relations in triples reconstructed via randomly replacing the entities and distinguishing datasets. Statistics of datasets used are shown.

FB15K and FB15K-237 are always regarded as two to learn embeddings by these mutually independent relations. 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.

Our scheme is readily incorporeal to any rule mining tool. And we choose AMIE+ (Galarraga 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:

\[
Q(h, r, t) = \|h + r - t\| + \alpha_1 \sum_{p \in P(h, t)} \sum_{\mu_i \in B(p)} \|C(p) - r\|
\]

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 thatFB15K-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

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

(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:

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

(5)

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

(6)

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

(7)

where the function \(\max(0, x)\) is defined to obtain the maximum value between 0 and \(x\), \(\gamma_1, \gamma_2, \gamma_3\) 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 \(\alpha_1, \alpha_2\) are two hyper-parameters respectively weight- ing the influence of paths and relation pairs embedding constraint. \(\beta\) 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 \(T\).

\[
T^- = (h', r, t) \cup (h', r', t) \cup (h, r, t')
\]

(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.
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 (Garcia-Duran 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 if 0.001, $\gamma_1$ and $\gamma_2$ 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 $\gamma_3$ in $\{1, 1.5, 2, 2.5, 3\}$, and the weight coefficients $\alpha_1$, $\alpha_2$ both in $\{0, 5, 1, 1.5, 2, 3, 5\}$. The best models are selected on validation sets. The resulting optimal of margin $\gamma_3$ and the weight coefficients $\alpha_1$, $\alpha_2$ 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 $\text{AthletePlaysInLeague}(x, y) \iff \text{AthleteIsTeam}(x, z) \land \text{TeamPlaysInLeague}(z, y)$ with confidence 0.96 will be used in path composition, but the rule $\text{PersonHasJobPosition}(x, y) \iff \text{HasSibling}(x, z) \land \text{PersonHasJobPosition}(z, y)$ 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.

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

| 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 | 72.8 |
| TransE | 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 7: Relation prediction results on FB15K. We use Hits@1 for better comparison because Hits@10 of all the models exceed 95%.

| Models | MR | Hits@1(%) |
|--------|----|-----------|
|        | 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 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...
Table 10: Ablation study by removing paths and length-2 rules as well as length-1 rules.

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

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 

$filmLanguage(x, y) \Leftarrow castactor(x, z) \land personLanguage(z, y)\quad (x, y) \Leftarrow castactor(z, z) \land personLanguage(z, y)$

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

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

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

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, M. H.; Stepanova, D.; Urbani, J.; and Weikum, G. 2016. Exception-enriched rule learning from knowledge graphs. In ISWC 2016, 234–251.

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

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

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

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

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

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

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

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, J.; Isele, R.; Jakob, M.; Anja Jentzsch, Dimitris Kontokostas, P. N. M. S. H. 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, 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, 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, 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, T.; tau Yih, W.; and Zweig, G. 2013. Linguistic regularities in continuous space word representations. In *NAACL-HLT 2013*, 746–751.

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

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

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

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

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