Compositional Learning of Relation Paths Embedding for Knowledge Base Completion

Xixun Lin¹, Yanchun Liang¹,², Renchu Guan¹,²∗

¹ Key Laboratory for Symbol Computation and Knowledge Engineering of National Education Ministry, College of Computer Science and Technology, Jilin University, Changchun 130012, China
² Zhuhai Laboratory of Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Zhuhai College of Jilin University, Zhuhai 519041, China

Abstract

Nowadays, large-scale knowledge bases containing billions of facts have reached impressive sizes; however, they are still far from completion. In addition, most existing methods only consider the direct links between entities, ignoring the vital impact about the semantic of relation paths. In this paper, we study the problem of how to better embed entities and relations into different low dimensional spaces. A compositional learning model of relation paths embedding (RPE) is proposed to take full advantage of additional semantic information expressed by relation paths. More specifically, using corresponding projection matrices, RPE can simultaneously embed entities into corresponding relation and path spaces. It is also suggested that type constraints could be extended from traditional relation-specific to the new proposed path-specific ones. Both of the two type constraints can be seamlessly incorporated into RPE and decrease the errors in prediction. Experiments are conducted on the benchmark datasets and the proposed model achieves significant and consistent improvements compared with the state-of-the-art algorithms for knowledge base completion.

1 Introduction

Large-scale knowledge bases (KBs) such as Freebase (Bollacker et al., 2008), Yago (Suchanek et al., 2007), NELL (Carlson et al., 2010) are critical to NLP applications, e.g., question answering (Dong et al., 2015), relation extraction (Riedel et al., 2013), and language modeling (Ahn et al., 2016). These KBs usually contain billions of facts and each fact is organized as the form of triple format (head entity, relation, tail entity), abbreviated as \((h, r, t)\), indicating that entities \(h\) and \(t\) hold the relationship \(r\).

However, even these KBs are impressively large, their coverages are still far from completion compared with real-world knowledge (Dong et al., 2014). Traditional approaches are suffered from the problem of feature sparsity and low-efficiency. Recently, encoding the entire knowledge base into a low-dimensional vector space to learn latent representations for entities and relations has been attracting widespread attention.

These knowledge embedding models yield better performance compared with prior works, in viewing of their low model complexity and high scalability. For example, TransE (Bordes et al., 2013) defines a score function \(S(h, r, t) = \|h + r - t\|\), to measure the plausibility for triples. It assumes that the score will get smaller if the triple \((h, r, t)\) is more likely to be correct, otherwise the score will get higher to indicate the triple \((h, r, t)\) is more likely to be incorrect. The latent representations for entities and relations are learned by optimizing a global margin-loss function based on the total plausibility of observed triples. TransH (Wang et al., 2014) proposes modeling each relation \(r\) as a hyperplane, head entity \(h\) and tail entity \(t\) are projected by hyperplane norm vector to better tackle the attributes of relation, i.e., 1-to-1 (Married to), 1-to-N (Has Children), N-to-1 (Gender is), and N-to-N (Friends to).

Both TransE and TransH assume that entities and relations embeddings are in the same embedding space. In TransR (Lin et al., 2015b), it defines a projection matrix for each relation \(r\), then the related entities are projected into different relation-specific embedding spaces. TransR considers the entities from multiple aspects and various relations on different aspects.

In spite of these knowledge embedding models...
are suitable to be deployed for large-scale KBs, most of them only exploit direct links connecting head and tail entities to predict potential relations between them. The new research direction using the semantic of relation paths to learn knowledge embeddings still needs to be explored (Lin et al., 2015a; Neelakantan et al., 2015; Guu et al., 2015; Toutanova et al., 2016).

In this paper, we propose a compositional learning model of relation paths embedding (RPE), utilizing the relation paths to improve the power of inference for more complicate situations. For example, the sequences of triples already existed in KBs, (J. K. Rowling, CreatedRole, Harry Potter), (Harry Potter, Describedin, Harry Potter and the Philosophers Stone) can be used to infer the new fact (J.K. Rowling, WroteBook, Harry Potter and the Philosophers Stone) which does not appear in original KBs. The motivation of RPE is that the semantic of relation paths which have been filtered by path ranking algorithm (PRA) should be similar to the semantic of relations between head and tail entities as much as possible. Based on this motivation we firstly propose the path-specific projection and path-specific type constraints to better learn the latent representations of entities and relations. In addition, we utilize PRA to pick up reliable relation paths also improve representation learning models interpretability, compared with pure data-driven mechanism knowledge embedding models used.

In RPE, we build upon the work of PTransE (Lin et al., 2015a). For each triple \((h,r,t)\), \(h,r\) will be employed by relation-specific and path-specific projections simultaneously. In relation-specific projection we define a matrix \(M_r\) for each relation \(r\); in path-specific projection we design two types of operation to dynamically construct \(M_p\) without extra parameters for completing this process. The new score function is defined as: 
\[
S(h,r,t) = \|M_r h + r - M_p t\| + \lambda \|M_p h + p^* - M_p t\|. 
\]

\(p^*\) denotes the relations composition for path representation. Figure 1 illustrates the basic idea for path-specific and relation-specific projections. We also suggest that based on the original motivation we can naturally extend the relation-specific to path-specific type constraints. Both type constraints can be seamlessly incorporated into our model. We evaluate RPE with the task of link prediction and triple classification on benchmark datasets. Experimented results demonstrate that our model achieves significant and consistent improvements compared with state-of-the-art results.

Our contributions can be summarized as: 1) To the best of our knowledge, we are firstly devise the path-specific projection to improve the power of more complicated inference for knowledge embedding models. It is also the first work that extends the relation-specific to path-specific type constraints to improve model’s discriminability. 2) In experiments, our model shows superior reasoning power to prior work including TransE, TransH, TransR and PTransE in link prediction and triple classification tasks on benchmark datasets.

In the remainder of this paper, we firstly introduce the most related knowledge embedding models, PTransE and PRA in Section 2; then we describe RPE formally in Section 3; experiments details are reported in Section 4; the conclusion and future work are discussed in the end.

2 Related Work

In this section, we firstly introduce some preliminaries about KBs. As mentioned in introduction, we denote a triple by \((h,r,t)\), KBs are organized as triple stores. All entities constitute the entity set \(\mathcal{E}\), all relations also constitute the relation set \(\mathcal{R}\). A relation path \(p\) can be represented as the sequence of relations, i.e., \(p=(r_1, r_2, \ldots, r_m)\). Bold lower case
letter \( v \) denotes a column vector, and bold upper case letter \( M \) denotes a matrix.

2.1 Knowledge Embedding Models

We firstly review four knowledge embedding models which only consider direct relations between entities. TransE considers that each entity or relation is a low dimensional vector in the same embedding space and the critical assumption for TransE is that every relation can be regarded as a translation from head entity to tail entity. For each triple \((h, r, t)\), TransE defines the score function as \( S(h, r, t) = ||h + r - t|| \). Obviously, this assumption is simple and it can not deal with the more complex relation attributes well, i.e., 1-to-N, N-to-1, and N-to-N.

To alleviate this fatal problem, TransH proposes setting different hyperplane normal vectors \( w_r \) for different relation \( r \), \((h, t)\) are projected to the hyperplane, denoting as \( h_h = h - w_r h w_r^T, t_t = t - w_r t w_r^T, (\text{Restrict } ||w_r||_2 = 1) \). The corresponding score function is \( S(h, r, t) = ||h_h + r - t_t|| \). Both TransE and TransH achieve translations in the same embedding space, TransR suggests each relation can be used to project entities into different relation-specific embedding spaces, in consideration of different relations may emphasis on different entity aspects. The projected entity vectors are \( h_h = M_h h, t_t = M_t t, M_r \) is relation-specific projection matrix. The new score function is correspondingly defined as \( S(h, r, t) = ||h_h + r - t_t|| \).

Another research direction focuses on using prior knowledge in format of relation-specific type constraints to improve the performance of prediction (Krompass et al., 2015; Chang et al., 2014; Wang et al., 2015). Noticing that each relation should possess Domain and Range fields to indicate the subject and object type respectively. For example, the relation haschildren’s Domain and Range types are both belong to person. Exploiting these limited rules can avoid the harmful influence of merely data-driven pattern, e.g., type-constrained TransE (Krompass et al., 2015) imposes these constraints on the global margin-loss ranking objective function to better distinguish similar entities or relations embeddings in embedding space.

These representative knowledge embedding models are befitting for applied on large-scale KBs. Unfortunately, all of them neglect the vital impact about semantic of relation paths, which means that they are disabled for complicated reasoning scenes.

2.2 PTransE and PRA

Large-scale KBs are very huge heterogeneous directed graphs, composed of entities as nodes and relations as different types of edges. They usually contain redundant relation paths, i.e., \( p_i (i=1, 2, \ldots, n) \) for concrete entity pair \((h, t)\), however, not all paths are meaningful for our reasoning. PTransE uses path ranking algorithm (PRA) to pick up reliable relation paths, more precisely, for each triple \((h, r, t)\), \( P_{all} = \{p_1, p_2, \ldots, p_k\} \) is the path set for entity pair \((h, t)\). PRA calculates \( P(t|h, p_i) \), the probability of reaching \( t \) from \( h \) following the sequences of relations indicated in \( p_i \), which can be recursively defined as:

If \( p_i \) is an empty path:

\[
P(t|h, p_i) = \begin{cases} 1 & \text{if } h = t \\ 0 & \text{else } h \neq t \end{cases} \tag{1}
\]

If \( p_i \) is not an empty path, \( p'_i \) is defined as \( r_1 \ldots r_{m-1} \)

\[
P(t|h, p_i) = \sum_{t' \in \text{Rand}(p'_i)} P(t'|h, p'_i) \cdot P(t', r_m) \tag{2}
\]

\( \text{Rand}(p'_i) \) is the set of target nodes where path \( p'_i \) ends at last.

PTransE only selects relation paths which their probability above a certain threshold \( \eta \) to obtain \( P_{filter} = \{p_1, p_2, \ldots, p_z\} \). It also explores three different compositions of relations for path representation \( p^* \). Experiment results demonstrate the ADD composition i.e., \( p^* = r_1 + r_2 + \ldots + r_z \) achieves the best performance. PTransE defines the new score function as:

\[
G(h, r, t) = S(h, r, t) + S(h, p, t) = ||h + r - t|| + \frac{1}{Z} \sum_{p_i \in P_{filter}} P(t|h, p_i) \cdot P(r|p_i) \cdot ||p^*_i - r|| \tag{3}
\]

where \( Z = \sum_{p_i \in P_{filter}} P(t|h, p_i) \) is normalization factor and \( P_i(r|p_i) = P_i(r, p_i) / P_i(p_i) \) is used to assist the calculation for relation paths reliance.

PRA uses the path-constrained random walk probabilities \( P(t|h, p) \) as path features to train different linear classifiers for corresponding relations, which is one of the most promising research for knowledge base completion also attracts much attention (Lao et al., 2011; Gardner et al., 2015; Wang et al., 2016; Nickel et al., 2016).
3 Our Model

All models mentioned above do not take full advantage of the semantic of relation paths, which has significant impact on learning meaningful latent representations for entities and relations. Here, we propose a novel compositional learning model of relation paths embedding (RPE) includes path-specific projection and path-specific type constraints tricks to utilize the semantic of relation paths to better overcome the problem of relations mapping properties and improve the discriminability.

3.1 Path-specific Projection

The key idea of RPE is the semantic of relation paths which have been filtered by PRA should be similar to the semantic of relation between entity pair. We suggest that for a triple \((h, r, t)\), RPE projects entity pairs vectors \(\mathbf{h}, \mathbf{t} \in \mathbb{R}^n\) in entity space to corresponding relation space and path space by projection matrices \(\mathbf{M}_r, \mathbf{M}_p \in \mathbb{R}^{m \times n}\) simultaneously (\(m\) is the dimension of relation embeddings, \(n\) is the dimension of entity embeddings, \(m\) is not required to be equal to \(n\)). The projected vectors \((\mathbf{h}_r, \mathbf{h}_p, \mathbf{t}_r, \mathbf{t}_p)\) in respective embedding spaces are denoted as:

\[
\begin{align*}
\mathbf{h}_r &= \mathbf{M}_r \mathbf{h} \\
\mathbf{h}_p &= \mathbf{M}_p \mathbf{h} \\
\mathbf{t}_r &= \mathbf{M}_r \mathbf{t} \\
\mathbf{t}_p &= \mathbf{M}_p \mathbf{t}
\end{align*}
\]

In view of relation paths are sequences of relations \(p=(r_1, r_2, \ldots, r_m)\), we dynamically use the \(\mathbf{M}_r\) to construct \(\mathbf{M}_p\) to decrease model complexity, we explore two operations for the formation of \(\mathbf{M}_p\), which are formulated as:

\[
\begin{align*}
\mathbf{M}_p &= \mathbf{M}_{r_1} + \mathbf{M}_{r_2} + \ldots + \mathbf{M}_{r_m} \quad \text{(ADD Operation)} \\
\mathbf{M}_p &= \mathbf{M}_{r_1} \ast \mathbf{M}_{r_2} \ast \ldots \ast \mathbf{M}_{r_m} \quad \text{(MUL Operation)}
\end{align*}
\]

ADD operation and MUL operation represent matrix addition and matrix multiplication, which are very simple and frequently be exploited in relevant works [GarciaDurn et al., 2015; Guu et al., 2015]. In our experiments, we evaluate these operations performance on two tasks of knowledge base completion (we named at PP-ADD and PP-MUL respectively). The new score function is defined as:

\[
G(h, r, t) = S(h, r, t) + \lambda \cdot S(h, p, t) = ||\mathbf{h}_r + \mathbf{r} - \mathbf{t}_r|| + \\
\frac{\lambda}{Z} \sum_{p_i \in P_{filter}} P(t|h, p_i) \cdot P_r(r|p_i) \cdot ||\mathbf{h}_p + \mathbf{p}_i^r - \mathbf{t}_p||
\]

(8)

For the composition of path representation \(\mathbf{p}^*\) we used the ADD composition, which as well shown in [Lin et al., 2015a; GarciaDurn et al., 2015]. \(\lambda\) is the hyper-parameter to balance the knowledge embedding score and relation path embedding score. In the experiments, we increase the limitation on these embeddings, i.e., \(||\mathbf{h}||_2 \leq 1, ||\mathbf{t}||_2 \leq 1, ||\mathbf{r}||_2 \leq 1, ||\mathbf{h}_r||_2 \leq 1, ||\mathbf{t}_r||_2 \leq 1, ||\mathbf{h}_p||_2 \leq 1, ||\mathbf{t}_p||_2 \leq 1\).

By exploiting the additional relation paths information RPE improves its flexibility when modeling the more complicate relation attributes. At the meantime, RPE can better tackle the more complex reason scenes compared with prior works.

3.2 Path-specific Type Constraints

In RPE, we assume the relation paths filtered by PRA should express similar semantics to relations between entity pair, so we can naturally extend the relation-specific type to path-specific type constraints. In type-constrained TransE, the distribution of the process for generate corrupted triples is uniform distribution.

In our model we borrow the idea from (Wang et al., 2014), incorporating the two type-constraints with Bernoulli distribution. For each relation \(r\), we denote the Domain\(_r\), Range\(_r\) to indicate the relation \(r\) subject and object types. \(\xi_{\text{Domain}_r}\) is the entity set which entities are conformed to Domain\(_r\), \(\xi_{\text{Range}_r}\) is the entity set which entities are conformed to Range\(_r\). We calculate the average number of tail entities for every head entity \(\text{teh}\) and the average number of head entities for every tail entity \(\text{het}\). We denoted the Bernoulli distribution with parameter \(\frac{\text{teh}}{\text{teh} + \text{het}}\) for each relation \(r\) incorporated with two type-constraints, which can be defined as: RPE samples entities from \(\xi_{\text{Domain}_r}\) to replace the head entity with the probability \(\frac{\text{teh}}{\text{teh} + \text{het}}\) and samples entities from \(\xi_{\text{Range}_r}\) to replace the tail entity with the probability \(\frac{\text{het}}{\text{teh} + \text{het}}\).

The objective function for our model is defined as:

\[
L = \sum_{(h, r, t) \in S} [L(h, r, t) + \frac{\lambda}{Z} \sum_{p_i \in P_{filter}} P(t|h, p_i) \cdot P_r(r|p_i) \cdot L(h, p_i, t)]
\]

(9)
\( L(h, r, t) \) is loss function for triples and \( L(h, p, i) \) is loss function for relation paths, which defined are:

\[
L(h, r, t) = \sum_{(h', r', t') \in S'} \max(0, S(h, r, t) + \gamma_1 - S(h', r', t'))
\]

\( (10) \)

\[
L(h, p, i) = \sum_{(h', r', t') \in S'} \max(0, S(h, p, i) + \gamma_2 - S(h', p, i', t'))
\]

\( (11) \)

We denote \( \mathcal{S} = \{(h_i, r_i, t_i) \mid i = 1, 2, \ldots, m\} \) is the set of all observed triples, \( \mathcal{S}' = \{(h'_i, r'_i, t'_i) \cup (h_i, r_i, t'_i) \mid i = 1, 2, \ldots, m\} \) is the set of corrupted triples, each element of \( \mathcal{S}' \) is obtained by randomly sampling from \( \zeta \). \( \mathcal{S}'' \) is the subset of \( \mathcal{S}' \), which each element is conformed to the two type-constraints with Bernoulli distribution. The \( \max(0, x) \) returns the maximum between 0 and \( x \). \( \gamma \) is the hyper-parameter of margin to separate corrected triples and corrupted triples. By exploiting the relation-specific and path-specific type constraints, RPE could better distinguish similar entities or relations embeddings in different embedding spaces, which can achieve better prediction quality.

### 3.3 Training Details

We adopted stochastic gradient descent (SGD) to minimize the objective function. We can use the method as shown in TransE to initiate entity and relation embeddings or use the embeddings that have been trained in RPE (initial) (we adopt its score function is

\[
G(h, r, t) = S(h, r, t) + \lambda \cdot S(h, p, t) = \|h + r - t\| + \lambda \sum_{p_i \in P_{filter}} P(t|h, p_i) \cdot P_r(r|p_i) \cdot \|h + p_i^r - t\|
\]

\( (12) \)

, we also evaluate it in our experiment as our baseline model). The projection matrices \( \mathbf{M} \) are initialized as identity matrices, projection vector are initialized as TransE suggested. RPE holds the local closed-world assumption (LCWA), each relations domain and range types are based on the instance level, collected by type information provided by KBs or the entities shown in our observed triples.

Noticing that each relation \( r \) owns its reverse relation \( r^{-1} \), therefore, to better learn the latent representations for entities and relations, RPE utilizes the reverse relation paths information, for example, the reverse relation path \( \text{LeBron James} \xrightarrow{\text{PlayFor}} \text{Cleveland Cavaliers} \xleftarrow{\text{BelongTo}} \text{NBA} \), the reverse relation path can be defined as \( \text{NBA} \xrightarrow{\text{BelongTo}^{-1}} \text{Cleveland Cavaliers} \xleftarrow{\text{PlayFor}^{-1}} \text{LeBron James} \).

To improve the computation efficiency, for every iteration we randomly sample a correct triple \((h, r, t)\), and the final score function for our model is defined as:

\[
F(h, r, t) = G(h, r, t) + G(t, r^{-1}, h)
\]

\( (13) \)

to void repetitive computation for \( G(t, r^{-1}, h) \). In our implementation, we restrict the path length is most at 3 in consideration of numerating all relation paths are time-consuming. Moreover, as the path-constrained random walk probability \( P(t|h, p) \) suggests, with the increase of path length, \( P(t|h, p) \) will get smaller, and it means the more likely the relation path will be filtered.

### 4 Experiments

#### 4.1 Datasets

We evaluate our model on two classical large-scale knowledge bases Freebase and WordNet. Freebase is a large collaborative knowledge base containing billions of facts about the real world, such as the triple (Beijing, LocatedIn, China) describes the fact that Beijing is located in China. WordNet is a large lexical knowledge base of English, each entity is a synset expressing a distinct concept. Each relationship is conceptual-semantic and lexical relations. We use two subsets of Freebase and one subset of WordNet, the FB15K, FB13 and WN11, which are provided by (Bordes et al., 2013) and (Socher et al., 2013) respectively. Table 1 gives the statistics of the datasets.

| Dataset | #Ent | #Rel | #Train | #Valid | #Test |
|---------|------|------|--------|--------|------|
| FB15K   | 14591| 1345 | 483142 | 50000  | 59071|
| FB13    | 75043| 13   | 316232 | 5908   | 23733|
| WN11    | 38696| 11   | 112581 | 2609   | 10544|

Table 1: The statistics of datasets.
level. In this case, for FB15K, we utilize the type information provided by \cite{xie-etal-2016}, as for FB13 and WN11, we do not depend on the auxiliary data, each relations domain and range type information is approximated by triples original dataset contains.

Our model is evaluated in two subtasks of knowledge base completion: link prediction \cite{bordes2013} and triple classification \cite{socher-etal-2013}.

4.2 Link Prediction

The task of link prediction is to predict the possible h or t for test triples (h,r,t) when h or t missing. We employ FB15K dataset for this task.

4.2.1 Evaluation Protocol

We follow the same evaluation procedures as used in \cite{bordes2013, weng2014, lin2015b}. Firstly, for each test triple (h,r,t), we replace the head entity h with every entities in \(\zeta\). Secondly, each corrupted triple is calculated by corresponding score function \(S(h,r,t)\). At last, we can rank of original correct entity with these scores in descending order.

Two metrics of evaluation are reported: the average rank of correct entities (Mean Rank) and the proportion of correct entities ranks in top 10 (Hits@10). Notice that if the corrupted triple already exist in knowledge base, it should not be considered as an incorrect triple. We prefer to remove these corrupted triples in our dataset, and we call this setting as Filter, if these corrupted are reserved, we call this setting as Raw. In both evaluation metrics, if the latent representations of entities and relations are better, the lower Mean Rank and the higher Hits@10 should be achieved.

4.2.2 Implementation

Because the FB15K is a benchmark data, we directly use the baseline results reported from \cite{lin2015b, lin2015}. Noticing that PTransE lists various variants, we report the best performance of PTransE (ADD, 2-step) in our results considering all evaluation metrics. We set the dimension of entity \(m\) and the dimension of relation \(n\) in the range of \(\{20, 50, 100, 120\}\). The margin \(\gamma_1\) for calculate the score of \(S(h,r,t)\) we set in the range of \(\{1, 2, 3, 4, 5\}\), and the margin \(\gamma_2\) for calculate the score of \(S(h,p,t)\) among \(\{3, 4, 5, 6, 7, 8\}\). The learning rate \(\alpha\) for SGD is set in the range of \(\{0.01, 0.005, 0.0025, 0.001, 0.0001\}\). The batch size \(B\) among \(\{20, 120, 480, 960, 1440, 4800\}\).

We use a grid search to determine the optimal parameters. The best configurations for RPE (PP-ADD) are \(n=100, m=100, \gamma_1=2, \gamma_2=5, \alpha=0.0001, B=4800, \lambda=1, \eta = 0.05\), we choose RPE (initial) to initial our model and take L1 norm for score function, and we traverse our models for 500 epochs. We exploit multi-thread training/testing to learn entity and relations distributed representations.

4.2.3 Results Analysis

Table 2 reports the results of link prediction, we denote RPE only with path-specific constraints as RPE (PC), and from the results we can observed that: 1) Our models significantly outperform the classical knowledge embedding models (TransE, TransH,TransR) and PTransE on FB15K with the metrics of mean rank and hits@10. The results demonstrate that the path-specific projection and path-specific type constraints can explore more semantics of relation paths, which are crucial for the task of large-scale knowledge base completion. 2) RPE only with the path-specific type constraints improve little performance compared with baseline models, we think it is mainly caused by RPE (PC) focuses on local information provided by related relations and entities, which certainly ignores some global information compared with the way of randomly select corrupted entities. RPE with path-specific projection achieves the best performance with 14.5% and 24.1% error reduction compare PTransE in raw and filter metrics respectively. In hits@10, RPE (PP-ADD) brings few

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|}
\hline
Metric & Mean Rank & Hits@10(%) \\
\hline
 & Raw & Filter & Raw & Filter \\
\hline
TransE & 243 & 125 & 34.9 & 47.1 \\
TransH (unif) & 211 & 84 & 42.5 & 58.5 \\
TransH (bern) & 212 & 87 & 47.7 & 57.4 \\
TransR (unif) & 226 & 78 & 43.8 & 56.5 \\
TransR (bern) & 198 & 77 & 48.2 & 68.7 \\
PTransE & 200 & 54 & 51.8 & 83.4 \\
RPE (initial) & 207 & 58 & 50.8 & 82.2 \\
RPE (PC) & 196 & 77 & 49.1 & 72.6 \\
RPE (PP-ADD) & 171 & 41 & 52.0 & \textbf{85.5} \\
RPE (PP-MUL) & 183 & 43 & 52.2 & 81.7 \\
RPE (PC + PP-ADD) & 184 & 42 & 51.1 & 84.2 \\
RPE (PC + PP-MUL) & 186 & 43 & 51.7 & 76.5 \\
\hline
\end{tabular}
\caption{Evaluation results on link prediction}
\end{table}
Table 3: Evaluation results on FB15K by mapping properties of relations. (%)

| Tasks                      | Predicting Head Entities (Hits@10) | Predicting Tail Entities(Hits@10) |
|----------------------------|-----------------------------------|-----------------------------------|
|                            | 1-to-1   | 1-to-N | N-to-1 | N-to-N   | 1-to-1   | 1-to-N | N-to-1 | N-to-N   |
| Relation Category          |         |        |        |          |         |        |        |          |
| TransE                     | 43.7     | 65.7   | 18.2   | 47.2     | 43.7     | 19.7   | 66.7   | 50.0     |
| TransH (unif)              | 66.7     | 81.7   | 30.2   | 57.4     | 63.7     | 30.1   | 83.2   | 60.8     |
| TransH (bern)              | 66.8     | 87.6   | 28.7   | 64.5     | 65.5     | 39.8   | 83.3   | 67.2     |
| TransR (unif)              | 76.9     | 77.9   | 38.1   | 66.9     | 76.2     | 38.4   | 76.2   | 69.1     |
| TransR (bern)              | 78.8     | 89.2   | 34.1   | 69.2     | 79.2     | 37.4   | 90.4   | 72.1     |
| PTransE                    | 91.0     | 92.8   | 60.9   | 83.8     | 91.2     | 74.0   | 88.9   | 86.4     |
| RPE (initial)              | 83.9     | 93.6   | 60.1   | 78.2     | 82.2     | 66.8   | 92.2   | 80.6     |
| RPE (PC)                   | 82.6     | 92.7   | 44.0   | 71.2     | 82.6     | 64.6   | 81.2   | 75.8     |
| RPE (PP-ADD)               | **92.5** | **96.6** | **63.7** | **87.9** | **92.5** | **79.1** | **95.1** | **90.8** |
| RPE (PP-MUL)               | 91.2     | 95.8   | 55.4   | 87.2     | 91.2     | 66.3   | 94.2   | 89.9     |
| RPE (PC + PP-ADD)          | 89.5     | 94.3   | 63.2   | 84.2     | 89.1     | 77.0   | 89.7   | 87.6     |
| RPE (PC + PP-MUL)          | 89.3     | 95.6   | 45.2   | 84.2     | 89.7     | 62.8   | 94.1   | 87.7     |

improvements. RPE with path-specific type constraints and path-specific projection is compromise between them.

Table 3 shows the evaluation results with separated types of relation properties on FB15K. From Table 3, we can conclude that 1) RPE (PP-ADD) outperforms all baseline models in all mapping properties of relations, in particular, for the 1-to-N,N-to-1,N-to-N types of relations, which plague knowledge embedding models, RPE (PP-ADD) improve 5.3%,6.0%,4.9% compared with previous state-of-the-art performances respectively. 2) RPE (PP-MUL) performs less than RPE (PP-ADD), and it is because RPEs relations path composition is not consistent with RPE (PP-MUL)s the composition of projection matrix for relation paths. Although RPE (path-constraint) improve little compared with PTransE, in the task of triple classification, we will indicate relation-specific and path-specific type constraints effectiveness. 3) We utilize the relation-specific projection matrix dynamically construct path-specific projection matrix, and entities are encoded into relation space by relation-specific projection matrix and path space by path-specific projection matrix simultaneously. Experimental results demonstrate our model possesses better expressivity when modeling more complicated inference scenarios and mapping properties of relations.

### 4.3 Triple Classification

We also conduct the task of triple classification on benchmark datasets to examine our models discriminative ability. Triple classification aims at predict a given triple \((h, r, t)\) is true or false.

#### 4.3.1 Evaluation Protocol

We select FB15K, FB13 and WN11 for this task. FB13 and WN11 already contain negative samples, as for FB15K we take the same process to randomly produce negative samples as (Socher et al., 2013) suggested. We set different relation-specific thresholds \(\{\delta_r\}\) to complete this task. For a test triple \((h, r, t)\) if its score \(S(h, r, t)\) below \(\delta_r\), we predict it as a positive triple, and otherwise as a negative triple. \(\{\delta_r\}\) is obtained by maximizing the classification accuracies on the valid set.

#### 4.3.2 Implementation

We also compare our model with prior works (Bordes et al., 2013; Wang et al., 2014; Lin et al., 2015b; Lin et al., 2015a). For WN11 and FB13 we directly use the results about knowledge embedding models reported in (Lin et al., 2015b). Because (Lin et al., 2015a) does not evaluate PTransE’s performance on this task, we use the code of PTransE released in (Lin et al., 2015a) to complete this part. The negative samples of FB15K are newly constructed to evaluate above models instead of directly using the reported results. The hyper-parameter intervals are the same as link prediction task. The best configurations for RPE (PC + PP-ADD) are: \(n=50, m=50, \gamma_1=5, \gamma_2=6, \alpha=0.0001, B=1440, \lambda=0.8, \eta =0.05\), taking \(L_1\) norm on WN11; \(n=100, m=100, \gamma_1=3, \gamma_2=6, \alpha=0.0001, B=960, \lambda=0.8, \eta =0.05\), taking \(L_1\) norm on FB13; \(n=100, m=100, \gamma_1=4, \gamma_2=5, \alpha=0.0001, B=4800, \lambda=1, \eta =0.05\), taking \(L_1\) norm on FB15K. We exploit RPE (initial) for initiation,
epochs are limited in 500 epochs.

### 4.3.3 Results AnAlysis

Table 4 lists the results of triple classification on different datasets. The results demonstrate that: 1) RPE (PC + PP-ADD) achieves the best performance on all datasets, which takes good advantage of incorporating path-specific type constraints and path-specific projection; 2) RPE (PC) improves the performance of RPE (initial), especially on FB15K, with 4.5%, 6.0%, 13.2% respectively. We consider lengthening the distances for similar entities in embedding space is required for specific problem. It also indicates that although LCWA can use the loss for type information and real relation-type information is predominant.

### 5 Conclusions and Future Work

In this paper, we propose a compositional learning model of relation paths embedding (RPE) for knowledge base completion. RPE aims at taking full advantage of additional semantics expressed by relation paths and making the semantic of relation paths which have been filtered by PRA similar to the semantic of relation between entity pair as much as possible. To the best of our knowledge, it is the first time to propose the path-specific projection which utilizes the relation-specific projection matrix to build path-specific projection matrices dynamically. Moreover, it also embeds the entities into relation space and path space to better deal with different type of relations, simultaneously. We firstly propose the path-specific type constraints to better distinguish similar entities in embedding space. In the future, we are going to 1) incorporate other potential semantic information into the relation paths modeling, such as the information provided by those intermediate nodes connected by relation paths; 2) explore relation path embedding in other applications associated with knowledge bases, such as knowledge base question and answering and distant supervision for relation extraction.

### Acknowledgments

The authors are grateful for the support of the National Natural Science Foundation of China (No. 61572228, No.61272207, No.61472158, No.61300147) and the Science Technology Development Project from Jilin Province (20140520070JH, 20160101247JC).

### References

[Bollacker et al.2008] Bollacker, Kurt and Evans, Colin and Paritosh, Praveen and Sturge, Tim and Taylor, Jamie 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, pages 1247-1250.

[Suchanek et al.2007] Suchanek, Fabian M and Kasneci, Gjergji and Weikum, Gerhard 2007. Yago: a core of semantic knowledge. In Proceedings of the 16th International Conference on World Wide Web, pages 697-706.

[Carlson et al.2010] Andrew Carlson and Justin Betteridge and Bryan Kisiel and Burr Settles and Estevam R. Hruschka and Tom M. Mitchell 2010. Toward an Architecture for Never-Ending Language Learning. In Proceedings of the 24th AAAI Conference on Artificial Intelligence, pages 1306-1313.

[Dong et al.2015] Li Dong and Furu Wei and Ming Zhou and Ke Xu 2015. Question Answering over Freebase with Multi-Column Convolutional Neural Networks. In Annual Meeting of the Association for Computational Linguistics (ACL).

[Riedel et al.2013] Sebastian Riedel and Limin Yao and Andrew McCallum and Benjamin M. Marlin 2013. Relation Extraction with Matrix Factorization and Universal Schemas. In NAACL, pages 74C84.

[Ahn et al.2016] Sungjin Ahn and Heeyoul Choi and Tanel Parnamaa and Yoshua Bengio 2016. A Neural Knowledge Language Model. CoRR, abs/1608.00318.

[Bordes et al.2013] Antoine Bordes and Nicolas Usunier and Alberto García-Durán and Jason Weston and Oksana Yakhnenko 2013. Translating Embeddings for Modeling Multi-relational Data. In NIPS.
[Wang et al.2014] Zhen Wang and Jianwen Zhang and Jianlin Feng and Zheng Chen 2014. Knowledge Graph Embedding by Translating on Hyperplanes. In AAAI.

[Lin et al.2015b] Yankai Lin and Zhiyuan Liu and Maosong Sun and Yang Liu and Xuan Zhu 2015. Learning Entity and Relation Embeddings for Knowledge Graph Completion. In AAAI.

[Neelakantan et al.2015] Arvind Neelakantan and Benjamin Roth and Andrew McCallum 2015. Compositional Vector Space Models for Knowledge Base Completion. In ACL.

[Guu et al.2015] Kelvin Guu and John Miller and Percy Liang 2015. Traversing Knowledge Graphs in Vector Space. In EMNLP.

[Toutanova et al.2016] Kristina Toutanova and Victoria Lin and Wen-tau Yih and Hoifung Poon and Chris Quirk 2016. Compositional Learning of Embeddings for Relation Paths in Knowledge Base and Text. In ACL.

[Krompass et al.2015] Denis Krompass and Stephan Baier and Volker Tresp 2015. Type-Constrained Representation Learning in Knowledge Graphs. CoRR, abs/1508.02593.

[Chang et al.2014] Kai-Wei Chang and Wen-tau Yih and Bishan Yang and Christopher Meek 2014. Typed Tensor Decomposition of Knowledge Bases for Relation Extraction. In EMNLP.

[Wang et al.2015] Quan Wang and Bin Wang and Li Guo 2015. Knowledge Base Completion Using Embeddings and Rules. In IJCAI.

[Lao et al.2011] Ni Lao and Tom M. Mitchell and William W. Cohen 2011. Random Walk Inference and Learning in A Large Scale Knowledge Base. In EMNLP.

[Gardner et al.2015] Matthew Gardner and Tom M. Mitchell 2015. Efficient and Expressive Knowledge Base Completion Using Subgraph Feature Extraction. In EMNLP.

[Wang et al.2016] Quan Wang and Jing Liu and Yuanfei Luo and Bin Wang and Chin-Yew Lin 2016. Knowledge Base Completion via Coupled Path Ranking. In ACL.

[Nickel et al.2016] Maximilian Nickel and Kevin Murphy and Volker Tresp and EvgeniyGabrilovich 2016. A Review of Relational Machine Learning for Knowledge Graphs, volume 104. In Proceedings of the IEEE, pages 11-33.

[GarcaDurn et al.2015] Alberto García-Durán and Antoine Bordes and Nicolas Usunier 2015. Composing Relationships with Translations. In EMNLP.

[Socher et al.2013] Richard Socher and Danqi Chen and Christopher D. Manning and Andrew Y. Ng 2013. Reasoning With Neural Tensor Networks for Knowledge Base Completion. In NIPS.

[Xie et al.2016] Ruobing Xie and Zhiyuan Liu and Maosong Sun 2016. Representation Learning of Knowledge Graphs with Hierarchical Types. In IJCAI.

[Dong et al.2014] Xin Dong and EvgeniyGabrilovich and Geremy Heitz and Wilko Horn and Ni Lao and Kevin Murphy and Thomas Strohmann and Shaohua Sun and Wei Zhang 2014. Knowledge vault: a web-scale approach to probabilistic knowledge fusion. In KDD.