Supplementary Material for
Learning to Exploit Long-term Relational Dependencies in Knowledge Graphs

A. Complementary Details

In this section, we introduce more details of the proposed framework. We first illustrate the architecture by an entity alignment example, and then give the algorithm of sampling relational paths with the biased random walks.

A.1. Architecture

Figure 5 shows the architecture of RSNs for the entity alignment task. It accepts two KGs as input and adopts an end-to-end framework to align the entities between them. Specifically, it first assembles the two KGs as a joint KG, and then repeatedly samples relational paths by the biased random walks on this KG. The generated paths are converted to embedding sequences according to the index of each element in the paths. It uses RSNs to model them and optimizes this process with type-based NCE. Finally, new alignment can be found by comparing the entity embeddings.

A.2. Algorithm of Biased Random Walk Sampling

We depict the algorithm of biased random walk sampling in Algorithm 1. It first precomputes the depth biases and the cross-KG biases to avoid repeated computation. Then, it samples the paths based on each triple instead of each entity, since using each entity for initialization may cause certain triples out of paths. It repeats the sampling process in terms of the sampling times and the maximal path length.

A.3. Implementation Details

We built RSNs based on the multi-layered LSTM (Hochreiter & Schmidhuber, 1997) (two layers for both entity alignment and KG completion) with Dropout (Srivastava et al., 2014). We conducted batch normalization (Ioffe & Szegedy, 2015) for both input and output of RSNs. KG embeddings and parameters of RSNs were initialized with Xavier initializer. We trained RSNs by Adam optimizer (Kingma & Ba, 2015) with mini-batches. Table 6 lists the hyper-parameter settings used in the experiments.

B. Entity Alignment Datasets

Random PageRank sampling is an efficient algorithm for large graph sampling (Leskovec & Faloutsos, 2006). It samples entities according to the PageRank weights and assigns higher biases to more valuable entities. However, it also favors high-degree entities. To fulfill our requirements on KG sampling, we first divided the entities in a KG into several groups by their degrees. Then, we separately performed random PageRank sampling for each group. The group number and size might be adjusted for several times to make the sampled datasets satisfying our requirements. To guarantee the distributions of the sampled datasets following the original KGs, we used the Kolmogorov-Smirnov (K-S) test to measure the difference. We set our expectation to $\epsilon = 5\%$ for all the datasets.

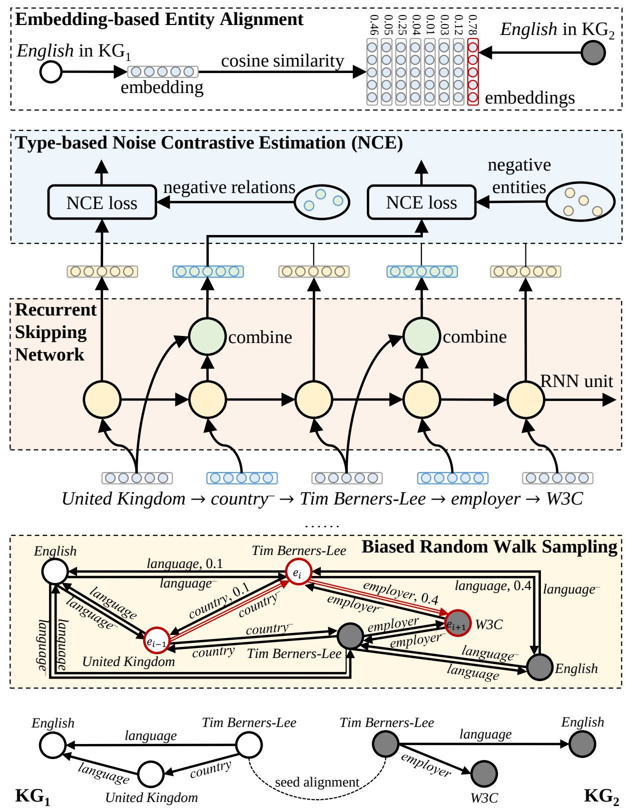


Figure 5. Architecture of the proposed method for entity alignment

The statistics of four couples of sampled datasets for entity alignment are shown in Table 7. For the normal datasets, they follow the degree distributions of the original KGs. For example, Figure 6 shows the degree distributions of DB-

Algorithm 1 Biased random walk sampling

```
Input: Triple set 𝒯, depth bias α, cross-KG bias β,
  sampling times n, max length l
Obtain biased transition probability matrices M_d, M_c;
for i := 1 to n do
  for each triple (s, r, o) ∈ 𝒯 do
    p := s → r → o
    repeat
      Look up M_d, M_c and compute normalized tran-
      sition probability distribution p_o of o;
      Sample next entity e from p_o;
      Sample a relation r' between o and e;
      p := p → r' → e;
    until length(p) ≥ l;
  end for
end for
```

Table 6. Experimental settings

	Entity alignment	KG completion
Embedding sizes	256	256
Batch sizes	512	2,048
Learning rates	0.003	0.0001
Bias hyper-parameters	$\alpha = 0.9, \beta = 0.9$	$\alpha = 0.7$
Path lengths	15	7

pedia and Wikidata, as well as the sampled datasets from different methods. We can see that our normal datasets best approximate the original KGs. For the dense datasets, we randomly removed entities with low degrees in the original KGs to make the average degree doubled, and then conducted the sampling. Therefore, the dense datasets are more similar to the datasets used by the existing methods (Chen et al., 2017; Sun et al., 2017; 2018; Wang et al., 2018).

C. More Experimental Analysis

C.1. KG Completion Results on FB15K-237

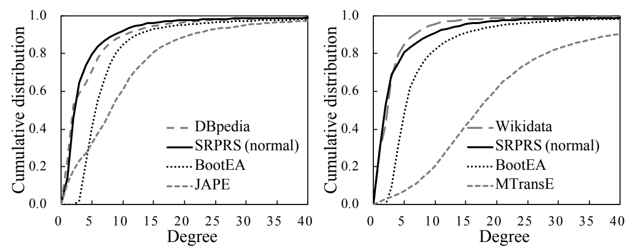
FB15K-237 (Toutanova & Chen, 2015) removes one side of symmetric relation pairs (e.g., *contains* versus *containedBy*). However, this may cut down the connectivity and cause unbalanced data distribution. For example, many methods achieve about 10% on Hits@1 for subject prediction, which is much lower than object prediction (about 30%). Thus, we argue that this dataset is still questionable. Furthermore, the test examples involving symmetric relations are just easy to be predicted, and we should not remove them due to the easiness. This may lean to the methods over-tailored to KG completion.

The experimental results on FB15K-237 are shown in Table 8. RotatE obtained the best results on this dataset, followed by ConvE and RSNs. It is worth noting that, while predicting the entities given two-thirds of one triple is not our primary goal, RSNs still achieved comparable or better

Table 7. Statistics of the entity alignment datasets

Datasets	Source KGs	Normal		Dense	
		#Rels.	#Triples	#Rels.	#Triples
DBP-WD	DBpedia (English)	253	38,421	220	68,598
	Wikidata (English)	144	40,159	135	75,465
DBP-YG	DBpedia (English)	219	33,571	206	71,257
	YAGO3 (English)	30	34,660	30	97,131
EN-FR	DBpedia (English)	221	36,508	217	71,929
	DBpedia (French)	177	33,532	174	66,760
EN-DE	DBpedia (English)	225	38,281	207	56,983
	DBpedia (German)	118	37,069	117	59,848

Each dataset contains about 15,000 entities.



(a) DBpedia (b) Wikidata

Figure 6. Comparison of degree distributions of the entity alignment datasets extracted by different methods

performance than many methods specifically focusing on KG completion. This revealed the potential of leveraging relational paths for learning KG embeddings.

C.2. Sensitivity to Proportion of Seed Alignment

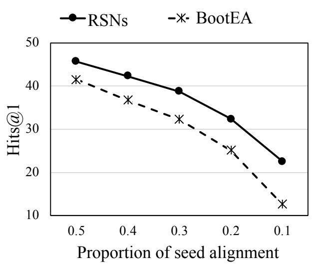
The proportion of seed alignment may significantly influence the performance of KG embedding methods. However, we may not obtain a large amount of seed alignment in real world. We assessed the performance of RSNs and BootEA (the best published method on the entity alignment task currently) in terms of the proportion of seed alignment from 50% down to 10% with step 10%.

We depict the results on the DBP-WD dataset in Figure 7. The performance of the two methods continually dropped with the decreasing proportion of seed alignment. However, the curves of RSNs are gentler than BootEA. Specifically, on the normal dataset, for the four proportion intervals, RSNs

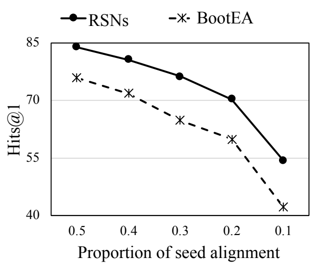
Table 8. KG completion results on FB15K-237

Methods	Hits@1	Hits@10	MRR
TransE‡	13.3	40.9	0.22
TransR‡	10.9	38.2	0.20
TransD‡	17.8	44.7	0.27
ComplEx	15.2	41.9	0.24
ConvE	23.9	49.1	0.31
RotatE	**24.1**	**53.3**	**0.34**
RSNs (w/o cross-KG bias)	20.2	45.3	0.28

lost 7.4%, 8.2%, 16.5% and 30.2% on Hits@1, respectively, while BootEA lost 11.8%, 12.0%, 22.3% and 49.8%. This demonstrated that RSNs are more stable. Additionally, when the proportion was down to 10%, the Hits@1 result of RSNs on the normal dataset is almost twice higher than that of BootEA, which indicated that modeling paths helps RSNs propagate the identity information across KGs more effectively and alleviates the dependence on the proportion of seed alignment.



(a) DBP-WD (normal)



(b) DBP-WD (dense)

Figure 7. Hits@1 results w.r.t. seed alignment