Learning to Exploit Long-term Relational Dependencies in Knowledge Graphs

南京大学万维网软件研究组 The Websoft Research Group, Nanjing University, China

Lingbing Guo, Zequn Sun, Wei Hu*

State Key Laboratory for Novel Software Technology, Nanjing University, China

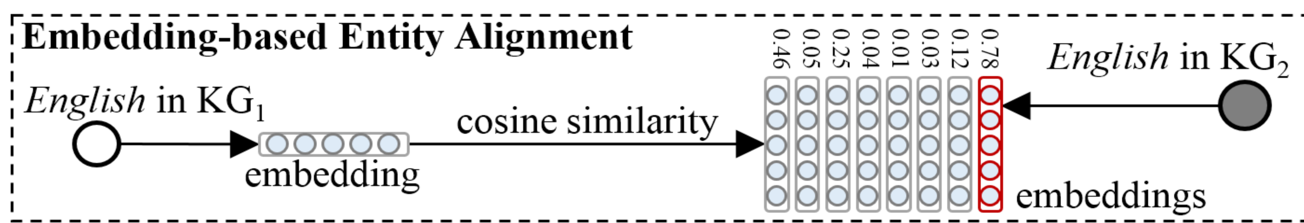
{lbguo, zqsun}.nju@gmail.com, whu@nju.edu.cn

Introduction

Knowledge graphs (KGs) store a wealth of structured facts about the real world. Each fact is structured in the form of (s, r, o), where s, o and r denote the **subject** entity, **object** entity and their **relation**, respectively

Oftentimes, a single KG is far from complete and cannot offer sufficient facts to applications. To solve this problem, two fundamental KG tasks are proposed

(i) Entity alignment, a.k.a. entity resolution or matching, which aims to integrate multiple KGs by identifying entities in different KGs referring to the same real-world object



(ii) KG completion, a.k.a. link prediction, which aims to complete the missing facts in a single KG, e.g., predict ? in (*Tim Berners-Lee*, *employer*, ?) or (?, *employer*, *W3C*)

The primary focus of many existing methods lies in learning from relational triples of entities. This so-called **triple-level learning** has two major limitations

- **Low expressiveness,** because it learns entity embeddings from a fairly local view (i.e., 1-hop relational neighbors)
- **Inefficient information propagation,** because it can only use relational triples to deliver semantic information within and across KGs

Recurrent Skipping Networks

We aim to learn from **relational paths**. A relational path is an **entity-relation chain**, where entities and relations appear alternately

United Kingdom → country⁻ → Tim Berners-Lee → employer → W3C

Recurrent neural networks (RNNs) are a popular class of neural networks which perform well on sequential data types. However, there still exist a few limitations

- A relational path has two different types: "entity" and "relation", which always appear in an alternating order
- A relational path is constituted by triples, but these basic structure units are overlooked by RNNs

Recurrent skipping networks (RSNs) can **shortcut** the current input entity to let it directly participate in predicting its object entity. Behind this simple skipping operation, there exists a new thought to adopt **tri-gram residual learning**

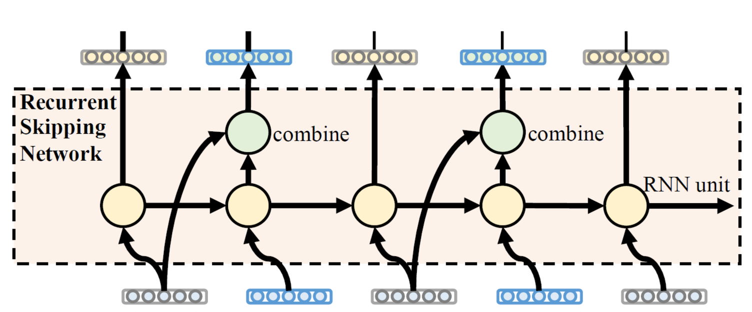


Table 1. Differences of RNNs, RRNs and RSNs, by an example (*United Kingdom*, *country*$^-$, *Tim Berners-Lee*, ***employer***, *W3C*)

Models	Optimize $F([\cdot], employer)$ as
RNNs	$F([\cdot], employer) := W3C$
RRNs	$F([\cdot], employer) := W3C - [\cdot]$
RSNs	$F([\cdot], employer) := W3C - Tim\ Berners\text{-}Lee$

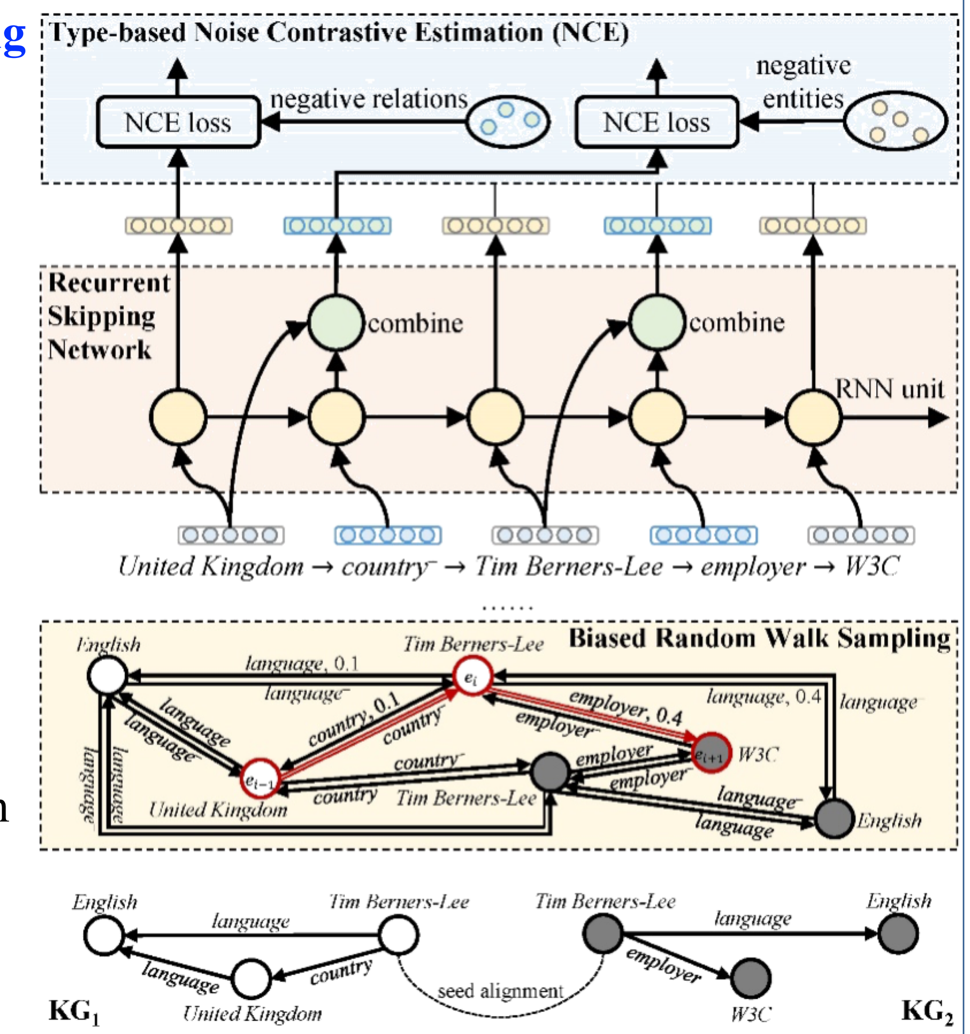
$[\cdot]$ denotes context (*United Kingdom*, *country*$^-$, *Tim Berners-Lee*)

As shown in the table, learning the residual between *W3C* and *Tim Berners-Lee* can make the optimization much easier. The skipping operation only increases a few more parameters, but provides an efficient way to remedy the major problem of leveraging sequence models to learn relational paths

Architecture

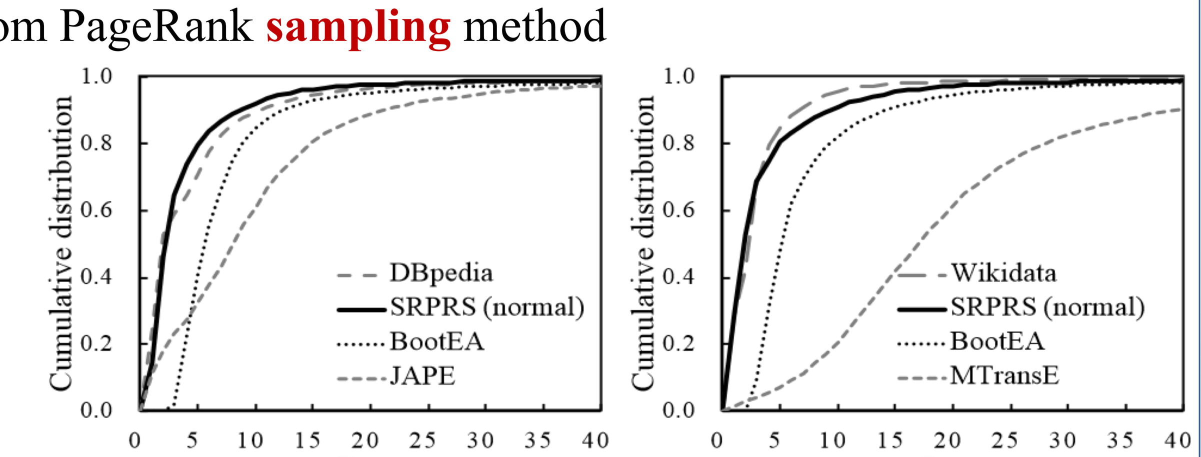
We present an **end-to-end framework** that leverages RSNs for entity alignment and KG completion. Three main modules are described as follows

- **Biased random walk sampling** generates deep and cross-KG relational paths
  - Deep paths carry **more relational dependencies** than triples to represent the relational roles of entities
  - Cross-KG paths serve as the **bridges** between two KGs to deliver alignment information
- **Recurrent skipping network** models relational paths to learn KG embeddings
- **Type-based noise contrastive estimation** evaluates the loss of RSNs in an optimized way



Experiments

Datasets used by the existing EA methods were sampled from real-world KGs, but their entity distributions are quite **different** from real ones. Thus, we design a segment-based random PageRank **sampling** method



Entity alignment results

RSNs performed best on all the datasets, especially for the datasets with normal entity distribution. Some KG completion methods such as ConvE [1] and RotatE [2] lost their advantages in this task. This may be due to that they were particularly designed for KG completion instead of embedding learning

Methods	DBP-WD			DBP-YG			EN-FR			EN-DE		
	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR
MTransE	22.3	50.1	0.32	24.6	54.0	0.34	25.1	55.1	0.35	31.2	58.6	0.40
IPTransE	23.1	51.7	0.33	22.7	50.0	0.32	25.5	55.7	0.36	31.3	59.2	0.41
JAPE	21.9	50.1	0.31	23.3	52.7	0.33	25.6	56.2	0.36	32.0	59.9	0.41
BootEA	32.3	63.1	0.42	31.3	62.5	0.42	31.3	62.9	0.42	44.2	70.1	0.53
GCN-Align	17.7	37.8	0.25	19.3	41.5	0.27	15.5	34.5	0.22	25.3	46.4	0.33
TransR†	5.2	16.9	0.09	2.9	10.3	0.06	3.6	10.5	0.06	5.2	14.3	0.09
TransD†	27.7	57.2	0.37	17.3	41.6	0.26	21.1	47.9	0.30	24.4	50.0	0.33
ConvE†	5.7	16.0	0.09	11.3	29.1	0.18	9.4	24.4	0.15	0.8	9.6	0.03
RotatE†	-	-	-	-	-	-	-	-	-	-	-	-
RSNs (w/o biases)	37.2	63.5	0.46	36.5	62.8	0.45	32.4	58.6	0.42	45.7	69.2	0.54
RSNs	**38.8**	**65.7**	**0.49**	**40.0**	**67.5**	**0.50**	**34.7**	**63.1**	**0.44**	**48.7**	**72.0**	**0.57**

"†" denotes KG completion methods conducted with the source code on the joint KGs. The same to the following.

KG completion results

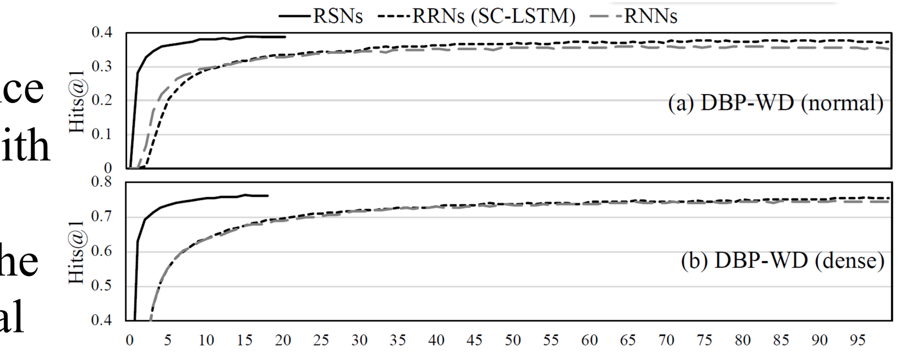
RSNs obtained comparable performance on both two datasets. It also performed better than all the translational models that also aim to learn KG embeddings rather than only complete KGs

Methods	Hits@1	Hits@10	MRR
TransE‡	30.5	73.7	0.46
TransR‡	37.7	76.7	0.52
TransD‡	31.5	69.1	0.44
ComplEx	59.9	84.0	0.69
ConvE	67.0	87.3	0.75
RotatE	**74.6**	**88.4**	**0.80**
RSNs (w/o cross-KG bias)	72.2	87.3	0.78

Methods	Hits@1	Hits@10	MRR
TransE‡	27.4	94.4	0.58
TransR‡	54.8	94.7	0.73
TransD‡	30.1	93.1	0.56
ComplEx	93.6	94.7	0.94
ConvE	93.5	95.5	0.94
RotatE	**94.4**	**95.9**	**0.95**
RSNs (w/o cross-KG bias)	92.2	95.3	0.94

Further Analysis

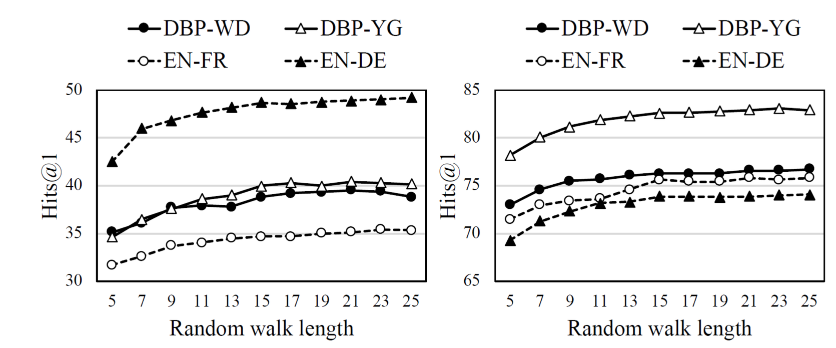
Comparing RSNs with RNNs



RSNs can achieve better performance with **only 1/30 epochs** compared with RSNs or RRNs (recurrent residual networks [3]), which demonstrated the effectiveness of the tri-gram residual learning in RSNs

Random walk length

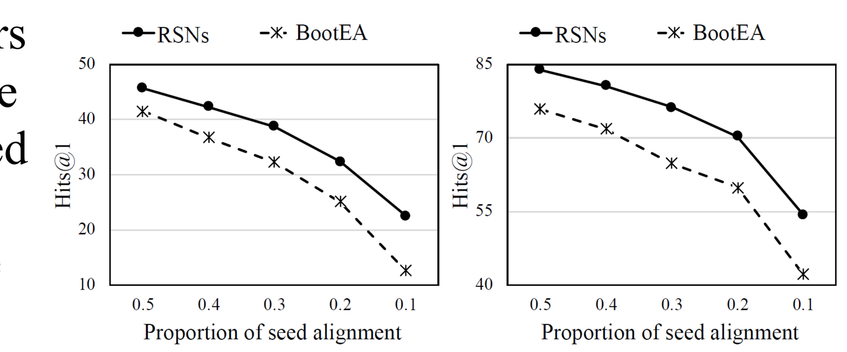
On all the datasets, the performance of RSNs increased steadily from length 5 to 15, which indicated that modeling **longer** relational paths can help KG embedding obtain better performance



(a) Normal datasets (b) Dense datasets

Proportion of seed alignment

Seed alignment refers to the entity pairs known ahead of time. The performance of the two methods continually dropped with the decreasing proportion of seed alignment. But the curves of RSNs are **gentler** than BootEA [4]. When the proportion was down to 10%, the hits@1 result of RSNs on the normal dataset was almost **twice higher** than that of BootEA.



(a) DBP-WD (normal) (b) DBP-WD (dense)

Conclusion

In this paper, we studied the **path-level KG embedding** learning

- We proposed **RSNs** to remedy the problems of using sequence models to learn relational paths
- We presented an **end-to-end framework**, which uses the biased random walks to sample desired paths and models them with RSNs
- Our experiments demonstrated that the proposed method can obtain superior performance for entity alignment and competitive results for KG completion

Future work includes studying a unified sequence model to learn KG embeddings using both relational paths and textual information

Code & datasets are available at https://github.com/nju-websoft/RSN

Main References

1. Dettmers, T., Minervini, P., Stenetorp, P., and Riedel, S. *Convolutional 2D knowledge graph embeddings*. In *AAAI*, pp. 1811–1818, 2018
2. Sun, Z., Deng, Z.-H., Nie, J.-Y., and Tang, J. *RotatE: Knowledge graph embedding by relational rotation in complex space*. In *ICLR*, 2019
3. Wang, Y. and Tian, F. *Recurrent residual learning for sequence classification*. In *EMNLP*, pp. 938–943, 2016
4. Sun, Z., Hu, W., Zhang, Q., and Qu, Y. *Bootstrapping entity alignment with knowledge graph embedding*. In *IJCAI*, pp. 4396–4402, 2018

Acknowledgements

This work is supported by the National Key R&D Program of China (No. 2018YFB1004300), the National Natural Science Foundation of China (No. 61872172) and the Key R&D Program of Jiangsu Science and Technology Department (No. BE2018131)