Rel4KC: A Reinforcement Learning Agent for Knowledge Graph Completion and Validation

Xiao Lin
AI Agents Group
DOCOMO Innovations Inc
Palo Alto, California, USA
xlin@docomoinnovations.com

Pero Subasic
AI Agents Group
DOCOMO Innovations Inc
Palo Alto, California, USA
psubasic@docomoinnovations.com

Hongfeng Yin
AI Agents Group
DOCOMO Innovations Inc
Palo Alto, California, USA
hyin@docomoinnovations.com

ABSTRACT

Reinforcement Learning (RL) has been recently adopted to train agents for knowledge graph completion tasks on structured database. However, new fact triples extracted through non-community contribution added to the database for completeness could be invalid due to noise in the input data and limitation of relationship discovery algorithm itself. In this study, we propose Rel4KC, a RL agent that learns from massive structured data and then performs completeness and correctness checking on the triple facts extracted from free text through neural relation extraction. Reward shaping based on embeddings of entities and relations are used to enhance RL agent's performance. The numerical experiments for a real-world problem demonstrate that the proposed approach yields promising results and has the potential of being adopted in knowledge graph generation and validation flow to ensure the trustworthiness of triple facts being populated into the knowledge base. Furthermore, a Japanese subset of the knowledge database is used to validate the multilingual extensibility of prototyped agent.

CCS CONCEPTS

• Computing methodologies → Artificial intelligence → Natural language processing, Knowledge representation and reasoning • Information systems → Information retrieval → Retrieval models and ranking • Computing methodologies → Machine learning → Learning paradigms → Reinforcement learning

KEYWORDS

Knowledge graph (KG), reinforcement learning (RL), Artificial intelligent (AI) agents, embedding, link prediction, entity, relation extraction

ACM Reference format:

Xiao Lin, Pero Subasic and Hongfeng Yin. 2019. Rel4KC: A Reinforcement Learning Agent for Knowledge Graph Completion and Validation. In *The 1st Workshop on Deep Reinforcement Learning for Knowledge Discovery* (DRL4KDD'19), August 5, 2019, Anchorage, AK, USA, 6 pages. https://doi.org/ 10.1145/1122445.1122456.

1. Introduction

Knowledge graph (KG) stores entities and corresponding relations in the format of fact triples (subject entity–relation–object entity, or e_s-r-e_o), and it has found broad applications in artificial intelligence (AI) for network architecture with logic reasoning capability (Battaglia et al. 2018), or to build explainable search and recommendation system (Zhao et al. 2019; Xie et al. 2018), social network representation (Perozzi, Al-Rfou and Skiena 2014), etc. Many natural language processing (NLP) tasks such as information retrieval, question-answering, chat-bot, machine reading also rely on a knowledge graph (Wang et al. 2017 & 2018; Yang and Mitchell 2017). Although there are several large-scale open domain knowledge bases available (Vrandecic and Krötzsch 2014; Färber et al. 2018), it is far from enough to cover whole humanity's knowledge span (Nguyen et al. 2018; Shang et al. 2019), and the KG itself does not possess logic reasoning capability (Socher, Chen, Manning and Ng 2013; Chen et al. 2018). Furthermore, evolution of human knowledge makes maintenance and coverage of a knowledge database very difficult. Therefore, much effort has been invested to extract knowledge from free text such as Wikipedia, news articles etc. to make the knowledge base complete (Subasic, Yin and Lin 2019). However, because the source of free text for such system is heterogeneous and noisy/error-prone, the fact triples extracted are not always correct. Besides, due to the consideration of a balance between knowledge completeness and query efficiency in this approach, the difficulty of automatic validation of extracted fact remains a challenging topic.

Accordingly, we develop Rel4KC, a reinforcement learning (RL) agent to validate triple facts extracted from free text. This RL agent can also function as link prediction for KG completion task. In other words, once the RL agent learns paths from existing

KG with large number of triples as training samples, not only it can be adopted in automation flow to validate the newly obtained triples from neural relation extraction, also it is capable of prediction of missing links from the KG itself. As illustrated in Figure 1, the RL agent interacts with a knowledge graph constructed from structured database (Wikidata), which is used as the training environment of this RL agent. Besides rewards evaluated by the predicted links in this database, the agent receives feedback with rewards evaluated on link prediction from embedding approach. The fact triples extracted from free text (Common Crawl) are then fed to the trained RL agent to determine their trustworthiness. If passing the validation, a triple is entered into target KG. If structured database is used as input, the RL agent predicts the missed links and to improve KG completeness.

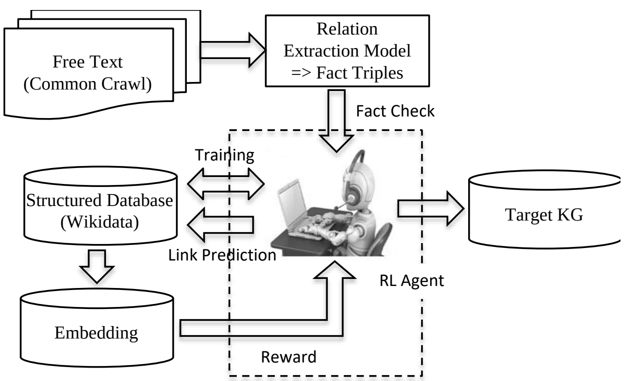

Figure 1: Flow Diagram of Deep Reinforcement Learning Agent for Knowledge Graph Checking & Completion

Our contribution in this study is that we are the first to architecture a system of adopting deep reinforcement learning agent for automation of the process of validating triples generated from relation extraction of free text. We conducted numerical experiments to demonstrate the effectiveness of RL agents in the KG construction flow, as well as its extensibility to multilingual applications. The system can be applied to large-scale knowledge graph construction. The code and data for this project will be released to public at: https://github.com/xlindii/Rel4KC.

2. Related Work

Reinforcement Learning method has been studied and applied actively in recent years to accomplish real world tasks (Chen et al. 2019; Dulac-Arnold, Mankowitz and Hester, 2019; Silver et al. 2018; Zhao et al. 2019). NLP applications include question answering (Buck et al. 2018; Wang et al 2018), relation extraction (Feng et al. 2018), text summarization (Paulus, Xiong and Socher 2018), neural machine translation (Wu et al. 2018), conversational agent (Dhingra et al. 2017) etc. Deep RL was first applied in DeepPath on finding relevant path of multiple hops between two entities in a knowledge graph for their similarity between random walk over nodes of a graph and Markov decision process (MDP) (Xiong, Hoang and Yang 2017). It showed that generally RL approach outperforms previous Path-Ranking Algorithm (PRA) as well as KG embedding method. Due to the computation complexity induced by huge action space of a given KG, DeepPath is mainly used for prediction of missed link between two entities. MINERVA (Das et al. 2018) is targeted to solve a more practical problem: to find the object entity given subject entity and relation (e_s-R -?). The approach is also featured by an entropy regularization added to boost choice of diverse paths from previous walks. KG is known to have very sparse connections which makes it hard for random walk in deep RL algorithm to receive positive rewards, which slows down the training performance. To overcome this difficulty, in addition to common formulation used by MINERVA, ReinforceWalk (Shen et al. 2018) emphasized its handling sparsity issue by adopting an RNN policy network combined with Monte Carlo Tree Search (MCTS). Reward sparsity issue is further studied by Lin, Socher and Xiong (2018), and in their approach they use existing embedding model to shape rewards, also perform hard walk stop through action dropout. This approach is also adopted by Godin and Kumar (2018) to achieve better performance.

It should be noted that recent deep RL agent studies are closely related with KG embeddings to achieve better accuracy and performance. Also, embedding methods are targeted for link prediction (Lin et al. 2015; Yang et al. 2015; Ristoski et al. 2019). The embeddings are low dimensional vectors meant to preserve the information in a KG. Besides modeling node entity as for common graph networks (Donnat et al. 2018; Hamilton et al. 2018), KG embeddings also need to model relations, either in separate vector spaces or sharing same one. Wang et al. (2017) conducted a comprehensive review of KG embedding approaches. In this study, we use four of these models: TransE, DistMult, ComplEx and ConvE. Inspired by word2vec models, TransE (Bordes et al. 2013) is the first model to introduce translation-based embedding assuming that relation and entity embeddings lie in the same vector space and relations are perceived as translation operations on entities. To overcome TransE matrix transformation complexity, DistMult (Yang et al. 2015) uses bilinear model and represents the relation as a diagonal matrix instead of full matrix, thus achieving significant performance scale-up. ComplEx (Trouillon et al. 2016) is derived from DistMult by only extending the embeddings from real values to complex floating numbers. It tends to accommodate modeling of asymmetric relations which is a limitation for its predecessor. Similar to ConvKB (Nguyen et al. 2018), ConvE (Dettmers et al. 2018) applies convolution layers on entity and relation instead of triples to nonlinear features. Thus, it is more computationally expensive, but outperforms significantly over the linear model. KG embedding remains an active research area and latest development includes embedding models such as CrossE that explicitly simulates crossover interactions (Zhang et al. 2019), Tucker decomposition of KG's tensor representation (), text enhanced modeling (Wang and Li 2016; Ding et al. 2018), etc.

Our deep RL agent is derived from Lin et al.'s model (2018) and applies KG embeddings to suppress false negative walks. The uniqueness of our study is that we apply reinforcement learning for knowledge graph construction, with a new metrics definition and feedback flow. The agent's action is not the end output but a key to validate a fact triple obtained from relation extraction from free text corpus. The system was built as a platform, so that adding customized knowledge database will help improve navigation for the random walk process.

3. Method

This section describes the system of knowledge graph validation using reinforcement learning. The problem can be described as: assuming entity set $\boldsymbol{\mathcal{E}} = \{e_i\}$ and relation set $\boldsymbol{\mathcal{R}} = \{r_i\}$ forms a knowledge graph $\boldsymbol{\mathcal{G}} = \{g_i = (e_{si} - r_i - e_{oi})\}$, where triple $g_i = (e_{si} - r_i - e_{oi})$ represents a relation fact r_i between subject entity e_{si} and object entity e_{si}, and given a new test triple $g_t = (e_{t1} - r_t - e_{t2})$ where $e_{t1}, e_{t2} \in \boldsymbol{\mathcal{E}}$ and $r_t \in \boldsymbol{\mathcal{R}}$, we need to validate $g_i \in \boldsymbol{\mathcal{G}} \Longrightarrow \mathbf{T}$, i.e., the fact is True (**T**). This is different from knowledge graph completeness problem, which targets to predict e_{t2} given e_{t1} and r_t. The problem is rendered as a random walk over graph $\boldsymbol{\mathcal{G}}$ to find a path starting from e_{t1} to reach node e_{t2} through a path $e_{t1} \dots e_i - r_i \dots - r_t - e_{t2}$ that can be naturally modeled as a Markov decision process.

3.1. Relation Extraction

As one of the major approaches to expand KG, relation extraction (RE) aims to extract relational facts between entities contained in text. Supervised learning approach is effective, but preparation of a high-quality labeled data is a major bottleneck in practice. One technique to avoid this is distant supervision (Riedel, Yao and McCallum 2010). Subasic at al. (Subasic, Yin and Lin 2019) propose a method to generate massive and high-quality relation extraction dataset through mapping entities in corpus sentences with the entities in an existing knowledge database. The approach obtains triples (subject entity–relation–object entity, or [e_s-r-e_o]) of facts from free text by first utilizing a structured database (Wikidata) to form a static KG through hierarchical traversal of links connected with domain keywords for compactness. This KG then is used to generate triples to train sequence tagging relation extraction model to infer new triples from free text corpus and generate a dynamic KG for completeness. The free text used in this study is Common Crawl corpus. Though large dataset is essential to achieve high accuracy, the model suffers false positives when generalized to arbitrary free text. Therefore, the training data is modified by adding extra negative samples for which entities are in matched sentence but not associated to its matching relation triples.

3.2. Reinforcement Learning Agent

During the random walk process of our RL agent on the graph $\boldsymbol{\mathcal{G}}$, at time moment i, it lands on node i and its state is defined by $s_i = [e_{t1}, e_{t2}, r_t, e_i]$, where e_{t2} is not observable to the agent at the moment. Current implementation treats the triples as reversible, meaning that if there is a triple enabling walk from e_1 to e_2, a walk from e_2 to e_1 is considered valid as well. The actions the agent can take are set of all the outgoing edges from node e_i: $A_s = \{(e_i \rightarrow r_i \rightarrow)\}$ where $r_i \in \mathcal{R}$. Since the relations are reversible, a possible action for an agent is that it could stay at the same node without moving forward and leaving the current node. The walk continues until preset maximum number of walks. Let the final state $s_i = [e_{t1}, e_{t2}, r_t, e_{t2}]$, the agent gets reward:

$$R_b(s_i) = \begin{cases} 1 \text{ if } (e_{t1}, r_t, e_{t2}) \in \mathcal{G} \\ 0 \qquad\qquad \text{otherwise} \end{cases} \tag{1}$$

It was observed by Lin, Socher and Xiong (2018) that in most situations false negatives will get rewards because KG by nature is incomplete. Therefore, the formulation is changed from hard reward to soft reward to improve performance by

$$R(s_t) = R_b(s_t) + (1 - R_b(s_t))\varphi(e_{t1}, r_t, e_{t2}) \tag{2}$$

where $\varphi(e_{t1}, r_t, e_{t2})$ is weight function of embedding for the corresponding triple. When $\varphi = 1$, Eq. (2) reduces to base reward indicating that walk lands on the correct node. Pretrained embedding (φ) itself is an approach to achieve link prediction, however it cannot be used to validate the new triple fact extracted from free text. In this study, we generated embeddings using three algorithms, DistMult, ComplEx, ConvE.

The policy is learned through a LSTM RNN network as diagramed in Figure 2.

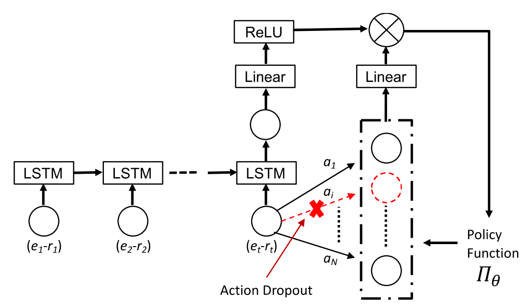

Figure 2. Schematics of Policy LSTM Network

The policy network is feedforward network with ReLU nonlinearity which takes in the current history representation (LSTM output from action and status) and the embedding for the query relation r_{t2} and outputs a probability distribution over the possible actions from which a discrete action is sampled. This is written as

$$\pi_\theta(a_t|s_t) = \sigma(A_t \times W_2 ReLU(W_1[e_{t1}; r_t; r_{t2}]) \tag{3}$$

where σ is softmax function. Optimization of policy network have several hyperparameters that can be tuned to achieve better results:

Number of multiple rollouts:	20
Moving window average:	1.0
Entropy regularization β:	0.02
Dimension of entity/relation: embeddings:	100
Number of LSTM layer:	3
Number of history steps:	400

4. Experiments and Results

4.1. Data Preparation

A knowledge graph used to prototype a soccer chatbot (Subasic, Yin and Lin 2019) was adopted in this study to validate the proposed RL agent flow. The knowledge graph is derived from full Wikidata database through hierarchical traversal of entities related with keyword "soccer". The graph is pruned to smaller size so that the computation can be completed on single Tesla V100 GPU. The obtained KG is summarized in Table 1. The column "Dynamic KG" represents a set of fact triples extracted from free text through neural RE approach, and any triples containing entity not defined in static KG are removed. The static KB is split into train/dev/test datasets by the ratio of 0.80:0.10:0.10 when training embeddings and RL agent.

	Static KG	Dynamic KG
Number of Entities	31,163	5,136
Number of Relations	168	69
Number of Triples	213,217	29,663

Table 1. Statistics of Static and Dynamic KG

The distribution of relation in this KG is highly biased, with most of the triples defined by few relations. In Figure 3, we show top 10 most frequent relations with total of 194,949 triples covering more than 90% of all triples.

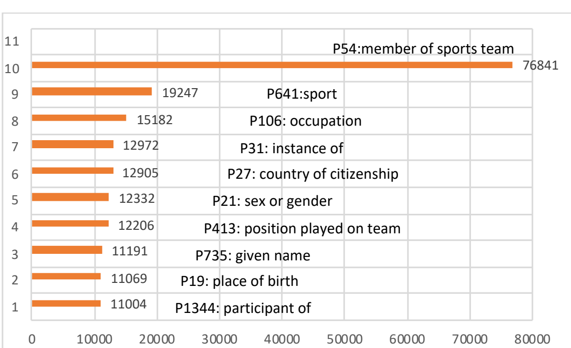

Figure 3. Top 10 Relations and Number of Triples Associated

The KG embeddings are trained using three approaches, and the result of hits@1/3/5/10 and MRR values is listed as in Table 2. Algorithm ConvE outperforms other approaches, therefore in the following tests, we use ConvE as default embedding generation unless there is an explicit indication of using other methods.

Embeddings	@1	@3	@5	@10	MRR
ConvE	**0.47**	**0.57**	**0.63**	**0.67**	**0.54**
ComplEx	0.47	0.56	0.60	0.66	0.53
DistMult	0.29	0.44	0.51	0.60	0.39

Table 2. Embedding of Static KG

4.2. Result

We use same policy optimization to minimize the loss function of reward and adopt most of the hyperparameters used by Lin et al. (Lin, Socher and Xiong 2018). To meet the limitation of GPU memory, we reduce the size of KG. Also, we use smaller embedding dimension, both for entity and relation, lowering from 200 to 100. Batch size can be reduced as well, but we see significant slow-down in training speed when the size is set to be as low as 128. We run two tests: one standard RL as a base line, and the other with reward shaping turned on. The obtained results are plotted in Figure 4. In the plot, the green bars represent base line values (without reward shaping). The difference values between reward shaping model and baseline is represented as red line segments on top of the green bars. As shown in the figure, reward shaping increases the accuracy of RL agent by up to 7.42 percent by absolute value. While the test shows RL's capability of self-learning the reasoning rules on the graph, it is observed that RL agent is unable to close a large gap with embeddings only approach. This calls for more tuning on the RL algorithm and its parameters.

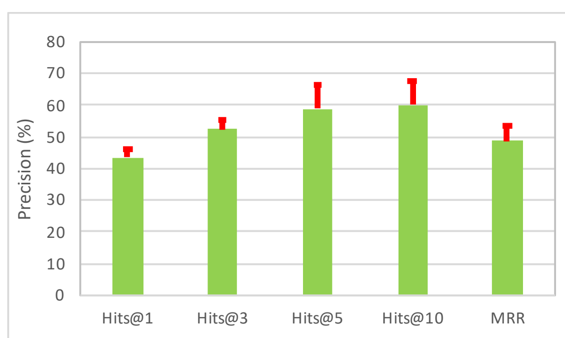

Figure 4. Base Line and Gain through Reward Shaping

Table 3 lists some examples where RL agent correctly identifies true fact triples. These facts are not in original structured database but extracted through relation extraction from free text. These triples are verified manually through online information. RL shows its capability to learn from structured database and perform reasoning from the relations between entities. On the other side, the test results show RL agent treats large portion of fact triples as negative. This might be caused by the effect of training RL agent using structured database as reward environment where knowledge graph by nature has sparse connections and is thus biased towards negative triples.

[Mehdi Taj] – [chairperson] – [Football Federation Islamic Republic of Iran]
[Algeria national football team] – [Yazid Mansouri] – [member of sports team]
[Argentina national football team] – [participating] – [1990 FIFA World Cup]
[Premier League Player of the Month] – [award received] – [Jürgen Klinsmann]

Table 3. Sample Fact Triples Rel4KC Identifies as True

4.3. Multilingual Support

We form a sub-graph by selecting entities and relations with Japanese labels, and perform numerical experiments as described in section 4.2. Instead of using Japanese dynamic graph extracted from Japanese text, the test suite dataset is used to mimic the input. Since entity ID and relation ID are used in the graph representation, the flow is actually language independent. The result is reported in Table 4 below. Because the sub-graph of Japanese is smaller than the original KG, the prediction accuracy is better compared to results in Table 2.

Number of Entity: 12,881 Number of Relation: 125 Number of Triples: 85,925	@1	@10	MRR
	0.49	0.69	0.56

Table 4. Performance of Japanese Version KG

SUMMARY

We proposed and demonstrated a flow to utilize a reinforcement learning (RL) agent trained to perform tasks of knowledge graph construction and validation. This RL agent is tested using a real-world knowledge graph designed for chatbot development. This RL agent identifies incorrect "facts" from relation extraction resulting from noise in text data. It can be used to improve KG completeness by predicting missed links on the existing structured database through reasoning.

Next, we want to improve the scalability of the proposed methodology and explore more advanced reinforcement learning algorithms for better performance. Another work could be to add more KG embedding models into the flow in order to explore the best options for the application.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their insightful comments to improve this work.

REFERENCES

[1] P. Battaglia et al., 2018, Relational Inductive Biases, Deep Learning, and Graph Networks, in *Google AI Research*: https://ai.google/research/pubs/pub47094.
[2] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, 2013, Translating Embeddings for Modeling Multi-relational Data, in *Proceedings of the 26th International Conference on Neural Information Processing Systems* (NIPS'13), pp. 2787–2795.
[3] C. Buck et al., 2018, Ask the Right Question: Active Question Reformulation with Reinforcement Learning, in *Proceedings of 6th International Conference on Learning Representations* (ICLR-18).
[4] M. Chen, A. Beutel, P. Covington, S. Jain, F. Belletti and E. Chi, 2019, Top-K Off-Policy Correction for a REINFORCE Recommender Systems, in *Proceedings of the 12th ACM International Conference on Web Search and Data Mining* (WSDM'19), pp. 456-464.
[5] W, Chen, W. Xiong, X. Fan and W. Wang, 2018, Variational Knowledge Graph Reasoning, in *Proceedings of NAACL-HLT* 2018, pages 1823–1832.
[6] R. Das, S. Dhuliawala, M. Zaheer, L. Vilnis, I. Durugkar, A. Krishnamurthy, A. Smola and A. MeCallum, 2018, Go for a Walk and Arrive at the Answer: Reasoning over Paths in Knowledge Bases using Reinforcement Learning, in *Proceedings of the 6th International Conference on Learning Representations* (ICLR 2018).
[7] T. Dettmers, P. Minervini, P. Stenetorp and S. Riedel, 2018, Convolutional 2D Knowledge Graph Embeddings, in *Proceedings of the 32nd AAAI Conference on Artificial Intelligence* (AAAI-18), pp. 1811-1818.
[8] B. Dhingra et al., 2017, Towards End-to-End Reinforcement Learning of Dialogue Agents for Information Access, in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics* (ACL'17), pp. 484–495.
[9] B. Ding, Q. Wang, B. Wang and L. Guo, 2018, Improving Knowledge Graph Embedding Using Simple Constraints, in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics* (ACL'18), pp. 110–121.
[10] C. Donnat, M. Zitnik, D. Hallac and J. Leskovec, 2018, Learning Structural Node Embeddings via Diffusion Wavelets, in *Proceedings of 24th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (KDD'18), pp. 1320-1329.
[11] M. Dubey, D. Banerjee, D. Chaudhuri and J. Lehmann, 2018, EARL: Joint Entity and Relation Linking for Question Answering over Knowledge Graphs, in *Proceedings of the 17th International Semantic Web Conference* (ISWC-18), vol. 1, pp. 108-126.
[12] G. Dulac-Arnold, D. Mankowitz and T. Hester, 2019, Challenges of Real-World Reinforcement Learning, to appear in *Proceedings of the 36th International Conference on Machine Learning* (ICML'19), 13 pages.
[13] M. Färber, B. Ell, C. Menne, A. Rettinger and F. Bartscherer, 2018, Linked Data Quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO, *Semantic Web Journal*, vol. 9, n0.1, pp. 77-129.
[14] J. Feng et al., 2018, Reinforcement Learning for Relation Classification from Noisy Data, in *Proceedings of the 32nd AAAI Conference on Artificial Intelligence* (AAAI-18), pp. 5779-5786.
[15] F. Godin and A. Kumar, 2018, Using Ternary Rewards to Reason over Knowledge Graphs with Deep Reinforcement Learning, in *Relational Representation Learning Workshop of the 32nd Conference on Neural Information Processing Systems* (NIPS 2018).
[16] W. Hamilton, P. Bajaj, M. Zitnik, Jurafsky and J. Leskovec, 2018, Embedding Logic Queries on Knowledge Graphs, in *Proceedings of the 32nd Conference on Neural Information Processing Systems* (NeurIPS 2018), pp 2030-2041.
[17] X. Lin, R. Socher and X. Xiong, 2018, Multi-Hop Knowledge Graph Reasoning with Reward Shaping, in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (EMNLP'18), pp. 3243–3253.
[18] Y. Lin, Z. Liu, M. Sun, Y. Liu and X. Zhu, 2015, Learning Entity and Relation Embeddings for Knowledge Graph Completion, in *Proceedings of the 29th AAAI Conference on Artificial Intelligence* (AAAI-15), pp. 2181-2187.
[19] D. Nguyen, T. Nguyen, D. Nguyen and D. Phung, 2018, A Novel Embedding Model for Knowledge Base Completion Based on Convolutional Neural Network, in *Proceedings of NAACL-HLT 2018*, pp. 327–333.

[20] R. Paulus, C. Xiong and R. Socher, 2018, A Deep Reinforced Model for Abstractive Summarization, in *Proceedings of the 6th International Conference on Learning Representations* (ICLR'18).

[21] B. Perozzi, R. Al-Rfou and S. Skiena, 2014, DeepWalk: Online Learning of Social Representations, in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (KDD'14), pp. 701-710.

[22] S. Riedel, L. Yao and A. McCallum, 2010, Modeling Relations and Their Mentions without Labeled Text. In: J. Balcázar, F. Bonchi, A. Gionis, M. Sebag (eds.) *Machine Learning and Knowledge Discovery in Databases*. ECML PKDD 2010. Lecture Notes in Computer Science, vol 6323. Springer, Berlin, Heidelberg.

[23] P. Ristoski, J. Rosati, T. Noia, R. Leone and H. Paulheim, 2019, RDF2Vec: RDF Graph Embeddings and Their Applications, *Semantic Web journal*, to appear. Available from http://www.semantic-web-journal.net/.

[24] C. Shang, Y. Tang, J. Huang, J. Bi, X. He and B. Zhou, 2019, End-to-End Structure-Aware Convolutional Networks for Knowledge Base Completion, in *Proceedings of the 33rd AAAI Conference on Artificial Intelligence* (AAAI-19).

[25] Y. Shen, J. Chen, P. Huang, Y. Guo and J. Gao, 2018, M-Walk: Learning to Walk over Graphs using Monte Carlo Tree Search, in *Proceedings of the 32nd Conference on Neural Information Processing Systems* (NeurIPS 2018).

[26] D. Silver et al., 2018, A General Reinforcement Learning Algorithm that Masters Chess, Shogi, and Go through Self-play, *Science*, vol. 362, no. 6419, pp.1140-1144.

[27] R. Socher, D. Chen, C. Manning and A. Ng, 2013, Reasoning with Neural Tensor Networks for Knowledge Base Completion, in *Proceedings of the 26th International Conference on Neural Information Processing Systems* (NIPS'13), pp. 926-934.

[28] P. Subasic, H. Yin and X. Lin, 2019, Building Knowledge Base through Deep Learning Relation Extraction and Wikidata, in *AAAI 2019 Spring Symposium on Combining Machine Learning with Knowledge Engineering* (AAAI-MAKE 2019).

[29] T. Trouillon, J. Welbl, S. Riedel, E. Gaussier and G. Bouchard, 2016, Complex Embeddings for Simple Link Prediction, in *Proceedings of the 33rd International Conference on Machine Learning* (ICML'16), JMLR, vol. 48, pp. 2071-2080.

[30] D. Vrandecic and M. Krötzsch, 2014, Wikidata: A Free Collaborative Knowledgebase, *Communications of the ACM*, 57(10):78-85.

[31] Q. Wang, Z. Mao, B. Wang and L. Guo, 2017, Knowledge Graph Embedding: A Survey of Approaches and Applications, *IEEE Transactions on Knowledge and Dara Engineering*, vol. 29, no. 12, pp. 2724-2743.

[32] S. Wang et al., 2018, R^3: Reinforced Ranker-Reader for Open-Domain Question Answering, in *Proceedings of the 32nd AAAI Conference on Artificial Intelligence* (AAAI-18), pp. 5981-5988.

[33] Z. Wang and J. Li, 2016, Text-Enhanced Representation Learning for Knowledge Graph, in *Proceedings of the 25th International Joint Conference on Artificial Intelligence* (IJCAI-16), pp. 1293-1299.

[34] L. Wu et al., 2018, A Study of Reinforcement Learning for Neural Machine Translation, in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (EMNLP'18), pp 3612–3621.

[35] W. Xiong, T. Hoang and W. Yang, 2017, DeepPath: A Reinforcement Learning Method for Knowledge Graph Reasoning, in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (EMNLP-17), pp. 564–573.

[36] Q. Xie, X. Ma, Z. Dai and E. Hovy, 2018, An Interpretable Knowledge Transfer Model for Knowledge Base Completion, in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics* (ACL-17), pp. 950–962.

[37] B. Yang, W. Yih, X. He, J. Gao and L. Deng, 2015, Embedding Entities and Relations for Learning and Inference in Knowledge Bases, in *Proceedings of International Conference on Learning Representations* (ICLR'15).

[38] B. Yang and T. Mitchell, 2017, Leveraging Knowledge Bases in LSTMs for Improving Machine Reading, In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pp 1436–1446.

[39] W. Zhang, B. Paudel, W. Zhang, A. Bernstein, and H. Chen, 2019, Interaction Embeddings for Prediction and Explanation in Knowledge Graphs, in *Proceedings of the 12th ACM International Conference on Web Search and Data Mining* (WSDM '19), 9 pages.

[40] X. Zhao, L. Xia, J. Tang and D. Yin, 2019, Deep Reinforcement Learning for Search, Recommendation, and Online Advertising: A Survey, in *ACM SIGWEB Newsletter* (SIGWEB'19 spring).