

Relation Prediction for Unseen-Entities Using Entity-Word Graphs

Yuki Tagawa, Motoki Taniguchi, Yasuhide Miura, Tomoki Taniguchi,
Tomoko Ohkuma, Takayuki Yamamoto, Keiichi Nemoto

Fuji Xerox Co., Ltd.

{tagawa.yuki taniguchi.motoki, yasuhide.miura, taniguchi.tomoki,
ohkuma.tomoko, yamamoto.takayuki, nemoto.keiichi}
@fujixerox.co.jp

Abstract

Knowledge graphs (KGs) are generally used for various NLP tasks. However, as KGs still miss some information, it is necessary to develop Knowledge Graph Completion (KGC) methods. Most KGC researches do not focus on the Out-of-KGs entities (Unseen-entities), we need a method that can predict the relation for the entity pairs containing Unseen-entities to automatically add new entities to the KGs. In this study, we focus on relation prediction and propose a method to learn entity representations via a graph structure that uses Seen-entities, Unseen-entities and words as nodes created from the descriptions of all entities. In the experiments, our method shows a significant improvement in the relation prediction for the entity pairs containing Unseen-entities.

1 Introduction

Knowledge graphs (KGs) are a set of triples in the form of (subject, relation, object), e.g., (Tokyo, Capital, Japan). These are an important resource for NLP tasks, such as entity linking (Radhakrishnan et al., 2018), question answering (Sun et al., 2018; Mohammed et al., 2018), and text generation (Koncel-Kedziorski et al., 2019).

Although many researches utilize KGs, these are still incomplete and miss some information. For example, in the Freebase (Bollacker et al., 2008), 71% of the person entities are missing a birthplace (Dong et al., 2014; Krompaß et al., 2015). In addition, as new information is increased with time, the KGs need to be updated. Therefore, to further increase the scale of the KGs, many researches have focused on Knowledge Graph Completion (KGC), which aims at predicting the missing information; for example, relation prediction that predicts the relation that holds between entities.

In recent years, embedding-based KGC methods have been proposed to learn entities and relations representations in KGs (Bordes et al., 2013; Xie et al., 2016b; Shi and Wenginger, 2017; Schlichtkrull et al., 2018; An et al., 2018; Nguyen et al., 2018, 2019). However, previous researches did not focus on the Out-of-KGs (OOKG) problem: OOKG is a problem that can not learn entity representations that are not included in the training triples (Unseen-entities). Therefore, the OOKG problem needs to be addressed to add new entities to the KGs such that they are automatically extended.

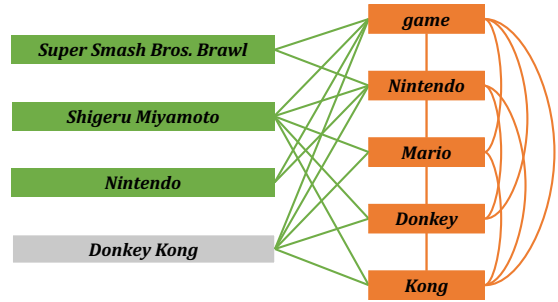
To cope with the OOKG problem, we build Unseen-entity representations using the entity descriptions. Figure 1(a) illustrates examples of entity descriptions, the descriptions include distinctive words (e.g., game, Nintendo) that represent the entity. Entities with co-occurring distinctive words in the descriptions are considered to be related. For example, the Unseen-entity “Donkey kong” was created by the “Shigeru Miyamoto”; therefore, these are highly relevant entities.

DKRL (Xie et al., 2016a) addresses the OOKG problem using entity descriptions. DKRL simply encodes the Unseen-entity descriptions separately with Convolutional Neural Networks (CNN) to obtain its representations. However, DKRL does not consider the relations with other entities while obtaining the Unseen-entity representations. By contrast, we can obtain better Unseen-entity representations by considering the information on related entities using word co-occurrences from the descriptions of all entities.

We address relation prediction and propose a method using entity descriptions to address the OOKG problem. Our method creates an Entity-Word graph from the descriptions of all entities. Figure 1(b) illustrates a part of the graph created from the entity descriptions. By creating

Super Smash Bros. Brawl
Super Smash Bros. Brawl, known in Japan as Dairantō Smash Brothers X, ... published by Nintendo for the Wii video game console. ... Brawl is the first game in the series to expand past Nintendo characters and ...
Shigeru Miyamoto
Shigeru Miyamoto is a Japanese video game designer and producer. ... Miyamoto joined Nintendo in 1977, ... Miyamoto has created include Mario , Donkey Kong , ...
Nintendo
Nintendo Co., Ltd. is a Japanese multinational consumer electronics company headquartered in Kyoto, Japan. Nintendo is the world's largest video game company by revenue. ...
Donkey Kong
Donkey Kong is a platform game developed in 1994 by Nintendo ... the player takes control of Mario and must rescue Pauline from Donkey Kong ...

(a) Entity descriptions



(b) A part of an Entity-Word graph

Figure 1: Examples of (a) entity descriptions and (b) a part of Entity-Word graph. The green node indicates a Seen-entity and the gray node indicates an Unseen-entity. The orange node indicates a word. The Entity-Word edge has a TF-IDF score and Word-Word edge has a PMI score.

the Entity-Word graph, even Unseen-entities can explicitly be connected with other entities. Our method encodes the graph with Graph Convolutional Networks (GCNs) (Kipf and Welling, 2017) to learn entity representations considering the global features of the entire graph. GCNs simplify the convolutional operations on the graph, and learn node representations based on their neighborhood information. GCNs are utilized for several NLP tasks (Zhang et al., 2018; De Cao et al., 2019). By encoding the Entity-Word graph with GCNs, not only the descriptions information but also information of the related entities is propagated to the Unseen-entities through words. We expect that the entity representations learned via our Entity-Word graph can contribute to the improvement in the performance of the KGC.

In summary, our contributions are as follows:

- We propose a method to learn Seen- and Unseen-entity representations using entity descriptions via GCNs. To the best of our knowledge, our work is the first consideration of utilizing entity descriptions via a graph structure to the KGC.
- In the experiments, our method significantly outperforms existing models at predicting the relation between the entity pairs containing Unseen-entities. Furthermore, our method outperforms existing models at predicting the relation between the pairs of Seen-entities.

2 Related work

TransE (Bordes et al., 2013) is a pioneering work on KGC. The energy function of TransE is defined

as: $E(s, r, o) = |s + r - o|$, where, s , r and o form the representation of a fact triple (s, r, o) . The embedding table in TransE converts a one-hot vector into a continuous vector space to learn entities and relations representations. TransE minimizes the loss function $L = \sum_{t \in S} \sum_{t' \in S'} \max(E(t) + \gamma - E(t'), 0)$, where, γ is the margin, S represents the fact triples in the KGs, and S' represents the unfact triples that are not in the KGs. The unfact triple t' is created by replacing the subject or object entity in the fact triple t with another one. Some variants of TransE are also proposed in this branch (Wang et al., 2014; Ji et al., 2015; Lin et al., 2015b,a). However, these models cannot learn Unseen-entity representations as these are not included in the training triples.

DKRL (Xie et al., 2016a) is an extension of TransE that uses entity descriptions to cope with the OOKG problem. While DKRL directly builds Unseen-entity representation from its description to avoid the OOKG problem, DKRL does not consider the relation between Unseen-entities and other entities. In this study, we construct an Entity-Word graph from the descriptions of all entities. By encoding the graph using GCNs, the entities information is also propagated to the Unseen-entities through words. However, DKRL does not have a mechanism by which the information of the related entities is propagated to the Unseen-entities.

There is a model that addresses triple classification for triples including Unseen-entities, which utilizes auxiliary knowledges instead of textual information to cope with the OOKG problem

(Hamaguchi et al., 2017). For example, to classify the triple (Blade-Runner, is-a, Science-fiction) containing the Unseen-entity ‘‘Blade-Runner’’ as either true or false, this model is provided with an auxiliary triple (Blade-Runner, based-on, Do-Androids-Dream-of-Electric-Sheep?). The authors deal with the special case and their problem setting is different from ours.

3 Proposed method

Our method first creates a TF-IDF, PMI, and Self-loop graph from the descriptions of all entities and then encodes each graph with different GCN layers to learn entity representations. Finally, our decoder predicts the relation between the entities using entity representations.

3.1 Graph creation

The TF-IDF graph has edges between entity and word nodes, and the weight of the edge has a TF-IDF score. On this graph, the Unseen-entities are explicitly connected to the other entities via word nodes. By encoding the TF-IDF graph with GCNs, our method can learn Unseen-entity representations considering the information on the related entities. The TF-IDF adjacency matrix A_{tf-idf} is expressed by the following formula:

$$A_{tf-idf}^{i,j} = \begin{cases} \text{TF-IDF}(i, j) (\text{entity } i, \text{word } j.) \\ \text{TF-IDF}(j, i) (\text{entity } j, \text{word } i.) \\ 0 (\text{otherwise.}) \end{cases}. \quad (1)$$

The PMI graph has edges between word nodes, and the weight of the edge has a PMI score. Word information is propagated among words by encoding this graph with GCNs. The PMI adjacency matrix A_{pmi} is expressed by the following formula:

$$A_{pmi}^{i,j} = \begin{cases} \max(\text{PMI}(i, j), 0) (\text{word } i, j.) \\ 0 (\text{otherwise.}) \end{cases}. \quad (2)$$

In addition to these graphs, we create a Self-loop graph with an edge from a node to itself. This connection indicates that the node representations are updated using their own representations when our method encodes this graph with GCNs. The Self-loop adjacency matrix A_{self} is a diagonal matrix in which all diagonal entries equal to 1.

3.2 Encoder

Our encoder is expressed by the following formulas:

$$h_{tf-idf}^{l+1} = \sigma(\tilde{A}_{tf-idf} h_{out}^l W_{tf-idf}^l), \quad (3)$$

$$h_{pmi}^{l+1} = \sigma(\tilde{A}_{pmi} h_{out}^l W_{pmi}^l), \quad (4)$$

$$h_{self}^{l+1} = \sigma(\tilde{A}_{self} h_{out}^l W_{self}^l), \quad (5)$$

$$h_{out}^{l+1} = h_{tf-idf}^{l+1} + h_{pmi}^{l+1} + h_{self}^{l+1}, \quad (6)$$

where h_{tf-idf} , h_{pmi} and h_{self} are the node representation matrices; h_{out} is an output node representation matrix; Each $\tilde{A} = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$ is a normalized adjacency matrix of each A , representing a graph, where D is the degree matrix of A ; W is a learning parameter matrix; and σ is an activation function. We used two GCN layers for the encoding. By aggregating the node representations and stacking multiple GCN layers, it is possible to learn the entity representations with higher-order neighborhood information.

3.3 Decoder

A decoder is expressed by the following formula:

$$p = \text{softmax}(\tanh(h_{out}^{subj} \oplus h_{out}^{obj}) W_{dec} + b), \quad (7)$$

where p is a probability distribution for the all relations, h_{out}^{subj} and h_{out}^{obj} are subject and object representations, respectively, \oplus indicates concatenation of vector representation, W_{dec} is a learning parameter matrix, and b is a bias term. We apply a cross entropy as a loss function.

4 Experiments

4.1 Datasets and evaluation metrics

In this study, we evaluated our method on a popular benchmark dataset FB15K (Bordes et al., 2013) from Freebase. Our method utilizes entity descriptions; therefore, similar to DKRL, we removed 47 entities that have no descriptions, and the triples containing these entities in FB15K. To evaluate the relation prediction for the pair including the Unseen-entity, we also evaluated our method on FB20K. Although FB20K shares the same training and validation triples as FB15K, the subject or object entity in the test triples is the Unseen-entity. The statistics of these datasets have been described in (Xie et al., 2016a).

Following previous researches on this branch, we used three metrics, namely Mean Rank (MR),

Pair type	Methods	MR	MRR	Hits@1
S-U [11,586]	DKRL (CNN) ³	15.99	0.69	62.8
	Our method	8.03	0.76	67.2
U-S [18,753]	DKRL (CNN)	12.72	0.71	64.8
	Our method	6.43	0.76	67.2
U-U [151]	DKRL (CNN)	25.66	0.33	7.9
	Our method	35.09	0.73	68.7
Total	DKRL (CNN)	8.91	0.74	65.8
	Our method	7.18	0.76	67.5

Table 1: Evaluation results on the FB20K (Unseen setting) and Filtered setting. S-U means that the object is an Unseen-entity. U-S means that subject is an Unseen-entity. U-U means that both are Unseen-entities. The number in the brackets indicates the number of samples. The best scores are highlighted in **bold**.

Mean Reciprocal Rank (MRR), and Hits@1 for evaluation: a lower MR, higher MRR, and higher Hits@1 indicate better performance. We also followed two evaluation settings, namely Raw (R) and Filtered (F) (Bordes et al., 2013).

4.2 Experimental settings

As a preprocessing for descriptions, we performed cleaning and tokenizing using (Kim, 2014) and then removed the stop words defined by the NLTK¹ and words less than 5 times in frequency. We calculated the TF-IDF and PMI values from the preprocessed descriptions, following which we created the Entity-Word graphs. The Entity-Word graphs contain the entities included in the training and test sets. Our model learns on the given the Entity-Word graphs and the gold labels of the training entity pairs. For our model², we selected the window size S as 5. In the 1st layer, we selected the dimension of W as 128 and ReLU as the activation function and then applied dropout to h_{tf-idf} , h_{pmi} , and h_{self} ; the ratio was 0.5. In the 2nd layer, we selected the dimension of W as 64 and an identity function as the activation function, and then applied L2 norm to h_{tf-idf} , h_{pmi} , and h_{self} . We selected one-hot node representations as the initial node representation matrix h_{out}^0 and applied Adam (Kingma and Ba, 2015) as an optimizer with the learning rate of 0.02.

¹<https://www.nltk.org/>

²We determined parameters by performing grid search for some candidates. We applied early-stopping on the loss on the validation set and selected a best model for the test time.

³For comparison and analysis, we implemented DKRL.

⁴We determined 0.5 as the weight of the DKRL(CNN) from the validation results.

Methods	MR		Hits@1	
	R	F	R	F
TransE (Bordes et al., 2013)	2.8	2.5	65.1	84.3
TransR (Lin et al., 2015b)	2.49	2.09	70.2	91.6
TKRL (RHE) (Xie et al., 2016b)	2.12	1.73	71.1	92.8
PTransE (ADD, 2step) (Lin et al., 2015a)	1.7	1.2	69.5	93.6
DKRL (CNN)+TransE ⁴	2.15	1.80	70.8	90.5
ProjE (wlistwise) (Shi and Weninger, 2017)	1.5	1.2	75.5	95.6
CACL (Oh et al., 2018)	1.52	1.19	77.6	96.4
Our method	1.51	1.13	72.6	95.0

Table 2: Evaluation results on the FB15K (Seen setting). The scores other than DKRL(CNN)+TransE are directly taken from (Oh et al., 2018) and the each original paper.

Entity1 : Bleach The DiamondDust Rebellion
Bleach: The DiamondDust Rebellion is the second animated film adaptation of the anime and manga series Bleach. ...
Entity2 : Cameron Arthur Clarke
Cameron Arthur ‘‘Cam’’ Clarke is a prolific American voice actor and singer, ...
Entity3 : Porco Rosso
Porco Rosso is a 1992 Japanese animated adventure film written and directed by Hayao Miyazaki . It is based on Hikōtei Jidai, a three-part watercolor manga by Miyazaki. ...

Table 3: Entities in the FB15K and their descriptions, and the relation ‘‘dubbing_performance/actor’’ holds between Entity1 and 2.

4.3 Evaluation results and discussion

In Table 1, our method shows a significant improvement on FB20K. Our method treats the Unseen-entities differently from DKRL. Entity-Word graphs are created and encoded in our method; therefore, our method can learn the representation of both the Seen- and Unseen-entities. Moreover, the information of the related entities is propagated to the Unseen-entities to assist the learning of Unseen-entities in our method. On the other hand, since DKRL only uses the descriptions information to obtain the Unseen-entity representations, DKRL does not consider the related entities information. Therefore, we concluded that our method outperforms the DKRL on the FB20K.

In Table 2⁵, our method shows a better MR than the previous models. Table 3 shows the entities in the FB15K and their descriptions, and the relation ‘‘dubbing_performance/actor’’ holds between Entity1 and 2. Our method correctly predicted this relation, whereas our implemented DKRL mispredicted this relation as ‘‘performance/actor’’. To accurately predict this relation, the KGC model needs to recognize the attribute that Entity1 is

⁵In many previous models, the MRR was not reported; therefore, we compare our method with the others based on the MR and Hits@1.

Japanese works and Entity2 is an American voice actor. However, the description of Entity1 does not directly indicate that Entity1 can be attributed to Japanese works. Therefore, the KGC model needs to recognize this from a few features such as the Japanese-English words “manga” and “anime”. The word “manga” also appears in the description of Entity3, which is a Japanese movie, and this description also contains words specific to Japanese works such as “Japanese animated” and “Hayao Miyazaki” who is famous Japanese animator. Our method makes use of the graph structure, Entity1 is located near the Japanese works entities such as Entity3 on the graph, which propagates this information to Entity1. Therefore, our method recognizes that Entity1 is Japanese works, and can correctly predict the relation.

5 Conclusion

In this study, we proposed a method to learn entity representations using entity descriptions via graph structure. In the experiments, the performance of the proposed model showed a significant improvement on the FB20K; furthermore, it outperforms the previous models on the FB15K. However, although the word order information (e.g., phrase) is an important clue for the relation prediction, our model disregards it when creating the Entity-Word graph. Thus, in future research, we plan to integrate our encoder with LSTM (Hochreiter and Schmidhuber, 1997) which can capture the word order information.

References

- Bo An, Bo Chen, Xianpei Han, and Le Sun. 2018. Accurate text-enhanced knowledge graph representation learning. In *Proceedings of NAACL*.
- Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of SIGMOD*.
- Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Proceedings of NIPS*.
- Nicola De Cao, Wilker Aziz, and Ivan Titov. 2019. Question answering by reasoning across documents with graph convolutional networks. In *Proceedings of NAACL*.
- Xin Dong, Evgeniy Gabilovich, Jeremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmman, Shaohua Sun, and Wei Zhang. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of SIGKDD*.
- Takuo Hamaguchi, Hidekazu Oiwa, Masashi Shimbo, and Yuji Matsumoto. 2017. Knowledge transfer for out-of-knowledge-base entities: A graph neural network approach. In *Proceedings of IJCAI*.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*.
- Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Knowledge graph embedding via dynamic mapping matrix. In *Proceedings of ACL*.
- Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of EMNLP*.
- Diederick P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of ICLR*.
- Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *Proceedings of ICLR*.
- Rik Koncel-Kedziorski, Dhanush Bekal, Yi Luan, Mirella Lapata, and Hannaneh Hajishirzi. 2019. Text Generation from Knowledge Graphs with Graph Transformers. In *Proceedings of NAACL*.
- Denis Krompaß, Stephan Baier, and Volker Tresp. 2015. Type-constrained representation learning in knowledge graphs. In *Proceedings of ISWC*.
- Yankai Lin, Zhiyuan Liu, Huan-Bo Luan, Maosong Sun, Siwei Rao, and Song Liu. 2015a. Modeling relation paths for representation learning of knowledge bases. In *Proceedings of EMNLP*.
- Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015b. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of AAAI*.
- Salman Mohammed, Peng Shi, and Jimmy Lin. 2018. Strong baselines for simple question answering over knowledge graphs with and without neural networks. In *Proceedings of NAACL*.
- Dai Quoc Nguyen, Tu Dinh Nguyen, Dat Quoc Nguyen, and Dinh Phung. 2018. A novel embedding model for knowledge base completion based on convolutional neural network. In *Proceedings of NAACL*.
- Dai Quoc Nguyen, Thanh Vu, Tu Dinh Nguyen, Dat Quoc Nguyen, and Dinh Phung. 2019. A capsule network-based embedding model for knowledge graph completion and search personalization. In *Proceedings of NAACL*.
- Byungkook Oh, Seungmin Seo, and Kyong-Ho Lee. 2018. Knowledge graph completion by context-aware convolutional learning with multi-hop neighborhoods. In *Proceedings of CIKM*.

- Priya Radhakrishnan, Partha Talukdar, and Vasudeva Varma. 2018. Elden: Improved entity linking using densified knowledge graphs. In *Proceedings of NAACL*.
- Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *Proceedings of ESWC*.
- Baoxu Shi and Tim Weninger. 2017. Proje: Embedding projection for knowledge graph completion. In *Proceedings of AAAI*.
- Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William Cohen. 2018. Open domain question answering using early fusion of knowledge bases and text. In *Proceedings of EMNLP*.
- Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of AAAI*.
- Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. 2016a. Representation learning of knowledge graphs with entity descriptions. In *Proceedings of AAAI*.
- Ruobing Xie, Zhiyuan Liu, and Maosong Sun. 2016b. Representation learning of knowledge graphs with hierarchical types. In *Proceedings of IJCAI*.
- Yuhao Zhang, Peng Qi, and Christopher D. Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. In *Proceedings of EMNLP*.