Embedding based Link Prediction for Knowledge Graph Completion

Russa Biswas
firstname.lastname@fiz-karlsruhe.de
FIZ Karlsruhe – Leibniz Institute for Information Infrastructure, Germany
Karlsruhe Institute of Technology, Institute AIFB, Germany

ABSTRACT

Knowledge Graphs (KGs) have recently gained attention for representing knowledge about a particular domain. Since its advent, the Linked Open Data (LOD) cloud has constantly been growing containing many KGs about many different domains such as government, scholarly data, biomedical domain, etc. Apart from facilitating the inter-connectivity of datasets in the LOD cloud, KGs have been used in a variety of machine learning and Natural Language Processing (NLP) based applications. However, the information present in the KGs are sparse and are often incomplete. Predicting the missing links between the entities is necessary to overcome this issue. Moreover, in the LOD cloud, information about the same entities is available in multiple KGs in different forms. But the information that these entities are the same across KGs is missing. The main focus of this thesis is to do Knowledge Graph Completion by tackling the link prediction tasks within a KG as well as across different KGs. To do so, the latent representation of KGs in a low dimensional vector space has been exploited to predict the missing information in order to complete the KGs.

KEYWORDS

Knowledge Graph Embedding, Encoder-Decoder Framework, Link Prediction, Entity Type Prediction, Entity Alignment

ACM Reference Format:
Russa Biswas. 2020. Embedding based Link Prediction for Knowledge Graph Completion . In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Knowledge Graphs (KGs) are large networks of real world entities and relationships between them. The facts are represented as a triple $< h, r, t >$, where h and t are the head and tail entities respectively and r represents the relation between them. Despite the huge amounts of relational data, one of the major challenges is that KGs are sparse and often incomplete as the links between the entities are missing. Furthermore, different KGs have information about the same real world entities but the fact that these entities in different KGs are same is missing.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Link Prediction (LP) is a fundamental task of Knowledge Graph Completion (KGC) that aims to estimate the likelihood of the existence of links between entities based on the current observed structure of the KG. LP task can be performed across different KGs to predict the missing links between two same entities across KGs and is also known as Entity Alignment. This thesis focuses on the KGC task based on predicting the missing links within the KG as well as across multiple KGs.

Moreover, most of the graph mining algorithms are proven to be of high complexity, deterring their usage in the application. Therefore, a necessity to learn the latent representation of a KG into a low dimensional space arises. To-date many algorithms are proposed to learn the embeddings of the entities and relations into the same vector space as mentioned in Section 2. However, none of the state-of-the-art (SOTA) models consider the contextual information of the KGs along with the textual entity descriptions to learn the latent representation of the entities and relations for the task of LP within the KG. This thesis focuses on proposing a model which takes the above described features into account and has been evaluated for the task of LP i.e., head, tail prediction as well as triple classification. On the other hand, due to the structural differences amongst multiple KGs, their embedding spaces also exhibit different characteristics. Therefore, for the entity alignment task, these different vector spaces generated for different KGs are to be aligned to a single space to predict the missing links between the same entities across different KGs.

2 STATE OF THE ART

In this section, the SOTA methods for Link Prediction and Entity Alignment have been discussed along with the research gaps.

Link Prediction. So far, different KG embedding techniques have been proposed which can be categorized as translation based models, semantic matching models, models incorporating entity types, models incorporating relation paths, models using logical rules, models with temporal information, models using graph structures, and models incorporating information represented in literals. The translational model [5] use scoring function based on distance and the translation is carried out with the help of a relation. TransE embeds entities and relations in the same embedding space which is created with the learning assumption $h + r \approx t$. GAKE [9] considers the contextual information by generating paths starting from an entity. A detailed description of these models for LP is provided in [7]. Another set of algorithms improve KG embeddings by taking into account different kinds of literals such as numeric, text or image literals. A detailed analysis of the methods is provided in [10]. Amongst them, DKRL [22] incorporates textual entity descriptions in the embedding model and uses TransE as the base model.

The textual entity descriptions present in the KGs provide information about the entity which might not be available otherwise in the KG. Also, the paths originating from an entity provide the structural contextual information about the neighboring entities. Therefore, in this thesis, paths and entity descriptions are modeled together to learn the embeddings of entities and relations for LP.

Entity Type Prediction. SDTyped [15] is a statistical heuristic link based type prediction mechanism. It exploits links between instances to infer their types using weighted voting. The model is based on the assumption that certain relations occur only with particular types. Another study [14] performs RDF type prediction in KGs with the help of the hierarchical SLCN algorithm using a set of incoming and outgoing relations as features for classification. In [13], the authors propose a supervised hierarchical SVM classification approach for DBpedia by exploiting the abstract and the categories of the Wikipedia articles. An embedding based model is proposed for entity typing [11] considering the structural information in the KG as well as the textual entity descriptions. A partially labelled attribute entity network is constructed containing structural, attribute and type information for entities and a deep neural network is employed to learn the representation of the entities. On the other hand, [21] proposes a multi-modal message passing network that learns end-to-end from the structure of KGs as wells as from multimodal node features such as text, numbers, images, geometrics, etc. The model uses neural encoders to learn embeddings for multi-modal node features which are projected to a unified embedding space along with the relational information.

On the contrary, in this thesis, different word embedding models such as Word2Vec, GloVe, FastText are used to generate embeddings for the entities and relations in a KG for the task of entity type prediction [3]. In this work, each triple is considered as a sentence and the entities and relations are considered as words. Also, the pre-trained vectors of RDF2Vec have been exploited for the task [19]. Furthermore, since the infobox types in Wikipedia are mapped to DBpedia entity types, different features from Wikipedia such as abstract, Table of Contents, Wikipedia categories have been used for the type prediction task [2, 4].

Entity Alignment. Entity Alignment is the task of aligning the same entities across different KGs. To do so, several embedding based methods have been proposed, in which a unified embedding space is learned using a set of already aligned entities and triples. A detailed description of these models for entity alignment is provided in [1]. The challenges of these models are: **(i)** They are supervised and require a set of aligned entities or triples as seeds for training. **(ii)** Some of the models require all the relations to be aligned between the KGs. However, in case of heterogeneous KGs which consist of different sets of relations, it is a challenging task to have a pre-aligned set of relations. **(iii)** The methods lack proper mechanisms to handle multi-valued relations. This thesis proposes an entity alignment model for heterogeneous KGs with multi-valued relations based on the unsupervised approach, i.e. without pre-aligned seeds for training.

3 RESEARCH QUESTIONS AND CONTRIBUTIONS

This section discusses the research questions and the corresponding contributions to address the challenges.

- *RQ1: Given an entity and a relation pair, how to predict the missing entity in a triple?*
  - The head or tail entity in a triple $< h, r, ? >$ or $<?, r, t >$ is predicted by defining a mapping function $\psi : E \times R \times E \rightarrow R$, where E and R are the set of entities and relations in the KG. A score is assigned to each triple where the higher the score of the triple indicates the more likely to be true.
- *RQ2: How to identify whether a given triple is valid or not?*
  - This is a triple classification task, in which a binary classifier is trained to identify whether a given triple is false (0) or true (1).
- *RQ3: How to predict the type information for an entity in a KG?*
  - Entity typing or Entity Classification is the process of assigning a type to an entity. To do so, different structural and literal information have been exploited to train a multi-label classification model for fine-grained entity typing.
- *RQ4: How to align the different embedding spaces of the KGs into a unified vector space to identify the* `owl:sameAs` *links?*
  - To align two different KG embedding spaces X and Y, a translation function τ coupled with a rotation function θ is introduced. The `owl:sameAs` links are then to be determined by vector similarity.

Therefore, the main contributions of this thesis are:

- A novel KG embedding model exploiting the structural as well as the textual entity descriptions in the KGs for head and tail prediction as well as triple classification.
- A neural network based multi-label hierarchical classification model for fine-grained entity typing using different features in the KG such as text and images along with the structural information.
- A novel translational model to align the different KG embedding spaces to identify the `owl:sameAs` links across multiple KGs.

4 KNOWLEDGE GRAPH COMPLETION

Link Prediction. This section depicts solution to research questions *RQ1 and RQ2* from Section 3. To encapsulate the contextual information, random walks of 4 hops are generated starting from each entity in the KG. Predicate Frequency Inverse Triple Frequency (PF-ITF) [16] is used to identify the important relations for each entity. The paths containing the most important relations for each entities is considered. A sequence-to-sequence (seq2seq) learning based encoder-decoder model [20] is adapted to learn the representation of the path vectors in the KGs as shown in Figure 1. Given a path sequence, which is a combination of entities and the relations between them, such as $\{e_1, r_1, e_2, r_2, ..., e_n\}$, the input to the encoder is the corresponding embeddings (computed using TransE). The entities and relations in the path are considered as words. These embeddings are passed through an attention based Bi-directional GRU

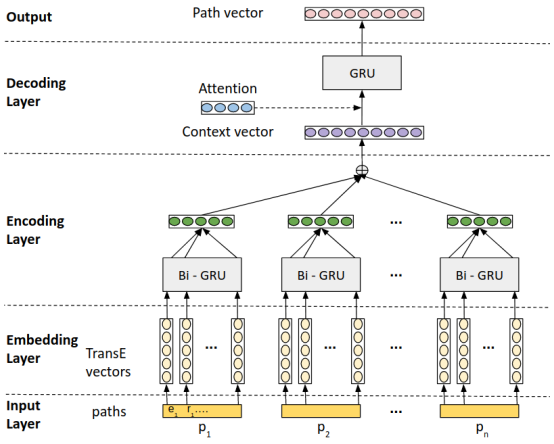

Figure 1: Encoder-Decoder Architecture

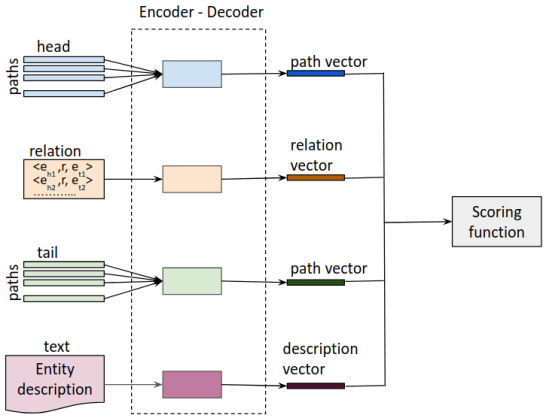

Figure 2: Overall Architecture

which encapsulates the information for all input elements and compresses them into a context vector which is then passed through the decoder. A scaled dot product is employed as the attention mechanism. The representation of the textual entity descriptions is obtained using SBERT [17], followed by the same encoder-decoder model. ConvE [8] is used as a scoring function for the head and tail prediction. The overall architecture is depicted in Figure 2. For triple classification, the vectors are passed through a Convolutional Neural Network (CNN). Triple classification as well as head and tail prediction of entity tasks are evaluated for FB15k and FB15k-237 datasets and the model outperforms the SOTA model DKRL as depicted in Table1. From the results, it can be inferred that the contextual information of the entities from the KGs. Also, SBERT model includes more semantic information resulting in better Link Prediction. Unlike DKRL, the proposed method is capable of handling shorter descriptions, i.e., descriptions less than or equal to three words. The proposed model has an improvement of 4.9% for Hits@1, 2.4% for Hits@3, and 4.2% for Hits@10 over DKRL for FB15k dataset. On the other hand, for FB15k-237 dataset, there is improvement of 4.2% for Hits@1, 1.9% for Hits@3, and 2.2% for Hits@10. Furthermore, as depicted in Table 2 the proposed model outperforms the SOTA models for triple classification as well.

Entity Type Prediction. In case of DBpedia, the Wikipedia infoboxes are the primary source of information. The types of the entities in Wikipedia infoboxes are mapped to the classes in DBpedia. Wikipedia infobox templates are created and assigned based on the categorical type of the article, i.e. articles belonging to a specific genre or type should be assigned the same template. The assignment of the infobox type to a Wikipedia article is executed based on the discussions between the contributors and the editors of the content of the Wikipedia article. However, no integrity tests are conducted to determine the correctness of the infobox assignment. This leads to incorrectness and incompleteness in the Wikipedia infobox type information which eventually leads to erroneous RDF type information in the KGs. Wikipedia Infobox Types have been predicted using the abstract, the text available in the Table of Contents (TOC) and the labels of Wikipedia Categories as features. In

Table 1: Results on LP with FB15k and FB15k-237 datasets.

Models	MR	MRR	Hits@1	Hits@3	Hits@10
FB15k					
DKRL	85.5	0.311	0.192	0.359	0.548
Our model (w/o Attn.)	87	0.316	0.222	0.365	0.5615
Our model (w Attn.)	**85**	**0.335**	**0.243**	**0.383**	**0.59**
FB15k-237					
DKRL	90.5	0.298	0.187	0.337	0.523
Our model (w/o Attn.)	90.5	0.314	0.217	0.349	0.527
Our model (w Attn.)	**90**	**0.316**	**0.229**	**0.356**	**0.545**

Table 2: Triple Classification

Models	FB15k	FB15k-237
TransE	79.8%	79.8%
TransH	79.9%	83.3%
TransR	82.1%	82.1%
TransD	88%	87.1%
Our model(w/o Att.)	**90.2%**	**89.2%**
Our model(with Att.)	**91.8%**	**90.9%**

this work, the Google pre-trained word vectors [1] of length 300 are used. For each Wikipedia article, an abstract vector, a TOC vector and a category vector are generated by performing average of the vectors of the words in the abstract. Two classifiers have been trained to predict the Wikipedia infobox types. The aforementioned document vector is used as the feature vector in the classification method using a Random Forest (RF) Classifier. For a multi-label Convolutional Neural Network (CNN), Wikipedia categories and TOC are considered as free text and sentence classification [12] method has been used where each Wikipedia article is considered

[1] https://code.google.com/archive/p/word2vec/

Table 3: Random Forest and CNN using micro F1 score

Features	RF	CNN
TOC	65.8%	76.5%
Abstract	86.4%	95.1%
Categories (C)	88.3%	**96.8%**
TOC + C	89%	**97.6%**
Abstract + TOC	88%	**96.1%**
Abstract + C	89%	**97.6%**
Abstract + TOC + C	89.7%	**98.3%**

Table 4: Accuracy of Vector Similarity and Neural Networks

Datasets	Vector Similarity	1D-CNN	FC-NN
59 classes, 500 entities/class	86.51%	81.78%	99.25%
86 classes, 2k entities/class	74.9%	53.67%	79.87%
81 classes, 4k entities/class	74.33%	53.49%	79.46%

as a sentence. 30 most popular infobox types with 5000 articles for each type from the Wikipedia 2016 version is used as dataset.

The experiments established the fact that the words in the categories of Wikipedia articles is well representative of the type of the article. With CNN, categories can predict the infobox types with a micro F_1-score of 96.8% which is 0.7% better than our previous results obtained when the prediction was based on the entire abstract and TOC combined as shown in Table 3. It has been shown in [2, 4] that word embeddings work better than the TF-IDF approach because the former captures contextual information of words.

Furthermore, a supervised as well as an unsupervised based approach have been explored to predict the entity types in DBpedia in [3, 19] exploiting the pre-trained KG embedding vectors generated from RDF2Vec [18]. The unsupervised approach is based on vector similarity approach. Since the entities of a class represent the characteristic of the class, therefore the average of the entity vectors in a class is considered as the Class Vector. Now to predict the type of a new entity, cosine similarity is calculated between the Class Vector and the test entity. On the the other hand, for the supervised approach, a 1D-CNN model is trained on the top of RDF2Vec vectors to predict the classes. Also, a Full Connected Neural Network (FC-NN) is used to predict the entity types as depicted in Table 4. The datasets used in this work are extracted from DBpedia and the results show that the FC-NN on top of RDF2Vec works the best for all the datasets. Entity Type Prediction task is referred to as *RQ3* in Section 3.

5 CONCLUSION AND FUTURE WORK

Experimental results show that the proposed model outperforms the textual description based embedding model DKRL for the link prediction task on both the datasets, FB15k and FB15k-237. Besides, on FB15k-237 dataset, the model outperforms several other SOTA models for both the tasks. The considerable improvement in the results as discussed earlier indicates that the contextual information plays an important role in the representation of the entities in the embedding space. Moreover, the attention based mechanism plays a vital role in representation of the entities and relations for both the tasks compared to the model without attention.

In future, to predict the missing links between the same entities across different KGs, entity alignment is to be done. The entity alignment task is yet to be addressed in this thesis. However, the basic idea is to adapt MUSE [6] which is an unsupervised multilingual word embedding alignment model to the KG alignment. A translation function coupled with a rotational function is to be used to align the related entities from different KGs. The same or related entities in different KGs will have overlapping information which could be exploited in an unsupervised manner.

ACKNOWLEDGEMENTS. This thesis is supervised by Prof. Dr. Harald Sack and Dr. Mehwish Alam.

REFERENCES

[1] Russa Biswas, Mehwish Alam, and Harald Sack. 2020. Is Aligning Embedding Spaces a Challenging Task? An Analysis of the Existing Methods. *arXiv preprint arXiv:2002.09247* (2020).
[2] Russa Biswas, Maria Koutraki, and Harald Sack. 2018. Predicting Wikipedia Infobox Type Information using Word Embeddings on Categories.. In *EKAW (Posters & Demos)*.
[3] Russa Biswas, Radina Sofronova, Mehwish Alam, and Harald Sack. 2020. Entity Type Prediction in Knowledge Graphs using Embeddings. *arXiv* (2020).
[4] Russa Biswas, Rima Türker, Farshad Bakhshandegan Moghaddam, Maria Koutraki, and Harald Sack. 2018. Wikipedia Infobox Type Prediction Using Embeddings.. In *DL4KGS@ ESWC*.
[5] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-Relational Data. In *NIPS*.
[6] Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word Translation Without Parallel Data. *arXiv preprint arXiv:1710.04087* (2017).
[7] Yuanfei Dai, Shiping Wang, Neal N Xiong, and Wenzhong Guo. 2020. A Survey on Knowledge Graph Embedding: Approaches, Applications and Benchmarks. *Electronics* (2020).
[8] Tim Dettmers, Minervini Pasquale, Stenetorp Pontus, and Sebastian Riedel. 2018. Convolutional 2D Knowledge Graph Embeddings. In *AAAI*.
[9] Jun Feng, Minlie Huang, Yang Yang, and Xiaoyan Zhu. 2016. GAKE: Graph Aware Knowledge Embedding. In *COLING*.
[10] Genet Asefa Gesese, Russa Biswas, Mehwish Alam, and Harald Sack. 2019. A Survey on Knowledge Graph Embeddings with Literals: Which model links better Literal-ly? *arXiv preprint arXiv:1910.12507* (2019).
[11] Hailong Jin, Lei Hou, Juanzi Li, and Tiansi Dong. 2018. Attributed and Predictive Entity Embedding for Fine-Grained Entity Typing in Knowledge Bases. In *Proceedings of the 27th International Conference on Computational Linguistics*.
[12] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *EMNLP*.
[13] Tomás Kliegr and Ondrej Zamazal. 2016. LHD 2.0: A Text Mining Approach to Typing Entities in Knowledge Graphs. *J. Web Sem.* (2016).
[14] A. Melo, H. Paulheim, and J. Völker. 2016. Type Prediction in RDF Knowledge Bases Using Hierarchical Multilabel Classification. In *WIMS*. 14.
[15] H. Paulheim and C. Bizer. 2013. Type Inference on Noisy RDF Data. In *ISWC*.
[16] Giuseppe Pirrò. 2015. Explaining and Suggesting Relatedness in Knowledge Graphs. In *ISWC*.
[17] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP-IJCNLP*.
[18] Petar Ristoski and Heiko Paulheim. 2016. Rdf2vec: Rdf graph embeddings for data mining. In *International Semantic Web Conference*.
[19] Radina Sofronova, Russa Biswas, Mehwish Alam, and Harald Sack. 2020. Entity Typing based on RDF2Vec using Supervised and Unsupervised Methods. In *To be published in Sattelite edition European Semantic Web Conference*.
[20] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*.
[21] WX Wilcke, P Bloem, V de Boer, RH van't Veer, and FAH van Harmelen. 2020. End-to-End Entity Classification on Multimodal Knowledge Graphs. *arXiv*.
[22] Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. 2016. Representation Learning of Knowledge Graphs with Entity Descriptions. In *AAAI*.