Improving Knowledge Graph Link Prediction Using Relation Extraction

Aditya Annavajjala, George Stoica, Chiraag Kaushik and Ketaki Jain

Friday 18th March, 2022

1 Summary

At the crux of many intelligent systems ranging from search engines to virtual assistants lies the ability to extract and form novel relationships between existing knowledge. **Knowledge Graph Link Prediction** (KGLP) and **Relation Extraction** (RE) are two closely intertwined tasks that investigate complementary functions of knowledge extraction. Link prediction is a supervised learning task over Knowledge Graphs (KG), which utilize a graphical structure to encode vast quantities of factual information as *(subject, relation, object)* triples. Given triples of this form, link prediction aims to predict an object given a subject and a relation. By contrast, relation extraction is a task which aims to predict the relation between a given subject and object pair, typically from a piece of text (i.e. independently of a knowledge graph structure).

In our project, we propose a multi-task learning framework for improving arbitrary KGLP methods by using the related task of RE to complement the training process. Specifically, we simultaneously train a RE method and a KGLP method while using the learned representations of the RE model to influence the KGLP method. We aim to explore whether the auxiliary information and implicit regularization provided by jointly training over a related task can improve performance.

To illustrate the efficacy of this multi-task paradigm, we first perform an extensive data processing step to align the entities (subjects and objects) and relations present in FB15K-237, a popular knowledge graph dataset, and the New York Times Annotated Corpus, which contains newspaper articles from the years 2005-2006. We then create a joint model, which uses FB15K-237 as the knowledge graph for link prediction and the NYT corpus as auxiliary data for relation extraction. In our experiments, we use a convolutional neural network (CNN) architecture for the link prediction task and a long-short term memory (LSTM) network for the relation extraction task. By training with a loss function that minimizes the error over both of these tasks (and the discrepancy between the predictions of the RE model and the inputs of the KGLP model), we demonstrate that the combined model achieves improved accuracy for link prediction when compared to the standalone single-task CNN model. We make the code available on GitHub.

2 Introduction

Knowledge can be defined as the ability to understand information about existing relationships and apply this understanding to uncover new relationships. We as humans inherently make

use of knowledge obtained from past experiences to reason and make decisions about the future. In machine learning systems, Knowledge Graphs (KGs) provide a concise way to store facts in the form of a directed graph. This structured format helps machines store related content as auxiliary information. KGs have widespread applications for virtual assistants, search and retrieval, online shopping recommendations, and social networks.

Knowledge graph link prediction (KGLP) is a learning task on KGs which helps identify edges that are likely to appear in the future if they do not exist already. The results of KGLP have diverse applications. At the corporate level, it can connect similar people, support team formation, and simplify collaboration. At the social level, it helps in maintaining census records across multiple locations, providing e-shopping recommendations and tailoring search engine results. More sensitive applications include anti-crime and counter-terrorism units, where potential links could be identified and their evolution forecasted. Naturally, KGLP methods have performances that are closely dependent on the data encoded in the KGs. However, while KGs contain an abundance of information, they are often automatically generated and incomplete [1]. Hence, training effective KGLP models remains a challenging problem. In the following, we explore how the paradigm of multi-task learning, using the tightly coupled task of relation extraction, can be used to effectively augment existing KGLP models for improved performance.

3 Related Work

The method we propose in this paper draws on prior work in 3 areas - Knowledge Graph Link Prediction, Relation Extraction and Joint Methods

- *Knowledge Graph Link Prediction:* KGLP methods aim to infer objects by mapping a given subject and a relation to an object set. For single hop methods, the object set consists of finite dimensional vectors, or embeddings. These are jointly transformed to produce an object set [2] [3] [4]. By contrast, multi-hop methods find connecting paths in the KG between subjects and objects to determine object sets [5, 6].

- *Relation Extraction:* Relation extraction methods take a subject and and object as input and aim to predict the relation between them. Given a sentence in the form of a sequence of tokens, RNNs [7, 8], CNNs [9], or transformer-based [10, 11] RE methods can infer relations. In addition to the sentence, graph-based methods use the structural characteristics of the sentence dependency tree to achieve strong performance.

- *Joint Methods* While coupling RE and KGLP methods has been previously explored[12, 13, 14], these approaches have focused on improving RE performance using KGLP. Notably, JRRELP[15] proposes jointly reasoning over three tasks - training an RE and KGLP model over their objectives from scratch, followed by training the composition of both models over the KGLP objective. This establishes a feedback connection between the KGLP and RE tasks, enhancing RE performance. In our project, we extend the idea proposed by JRRELP by performing the reverse: improving arbitrary KGLP methods using existing RE models.

4 Technical Details

4.1 Data Processing

In this section, we describe our chosen datasets and outline the technical steps and challenges while preparing our data for training the joint model. For link prediction, we use the a subset of the popular Freebase knowledge graph [16] called FB15K-237, which contains a total of 93372 triples in the training set, 12072 in the validation set, and 13709 in the test set. This knowledge graph is particularly well-suited to our task since contains a large amount of missing data, which is precisely the regime in which we expect a multi-task framework to be of use. We additionally use the New York Times Annotated Corpus [17] from the years 2005-2006 as a textual corpus for training a relation extraction model.

Dataset alignment: In order to run a joint model between KGLP and RE tasks, we require the datasets have aligned entity and relation identifiers and that there is an overlap between the entities and relations present in the two datasets. We start with the preprocessed data used in [13]. This data contained aligned identifiers between NYT and a subset of Freebase called FB15K (a strictly larger subset than FB15K-237). As a first step, we use the CoreNLP [18] package to tokenize the sentences from the NYT corpus and perform named-entity recognition (NER) to extract and classify the entities in each sentence. This process required careful attention to ensure that the subject/object of each sentence matched the form expected by the knowledge graph. Specifically, we had to account for subjects/objects which were combined into one word, words with inconsistent capitalizations, and inconsistent lengths between the NERs and sentence tokens (e.g. due to punctuation tokens). As a final step, since we choose to use the smaller knowledge graph set FB15K-237, we filtered all NYT triples to only keep those with corresponding object and relation in FB15K-237.

4.2 Joint Model

Before presenting our main method, we describe our various learning problems and approaches used at a more fine-grained level. The remainder of this section consists of the following: (1) Link Prediction, (2) Relation Extraction, and finally (3) Merging the two together.

4.2.1 Knowledge Graph Link Prediction

The objective of Knowledge Graph Link Prediction (KGLP) is to infer a set of objects T given a question of the form "entity-relation-?." It is assumed that the correct object(s) are existing nodes in the KG. Figure 1 illustrates this task. Let e_s, r and e_t denote one-hot encoded representations of the subject, relation and object of a KG triple. Let $E = \{e_i\}_1^{N_e}$ and $R = \{r_j\}_1^{N_r}$ denote the set of entities and relations in the KG with N_e and N_r demarcating the number of nodes and relations respectively. We then define the embedding matrices $\mathbf{E} \in \mathbb{R}^{N_e \times d}$ and $\mathbf{R} \in \mathbb{R}^{N_r \times d}$ to store the associated embeddings for entities and relations respectively. Using these matrices, we can characterize many prominent KGLP methods by

the following abstract model,

$$\mathbf{s} = \mathbf{E}e_s, \mathbf{r} = \mathbf{R}r \qquad \texttt{Embedding Lookup} \qquad (1)$$

$$\hat{\mathbf{e_t}} = f(\mathbf{s}, \mathbf{r}) \qquad \texttt{Object Prediction} \qquad (2)$$

$$p(O|e_s, r) = \sigma(\mathbf{E}\hat{\mathbf{e_t}} + \mathbf{b}) \qquad \texttt{Probability Estimation} \qquad (3)$$

Where $\hat{\mathbf{e_t}}$ the embedding representation for a set of correct object entities O, and the "Probability Estimation" step computes the elementwise probability that a model assigns to every individual entity being in O.

ConvE and KGLP Loss. We define ConvE [19] within the above framework by simply specifying the "Object Prediction" equation $\hat{\mathbf{e_t}} = \text{Conv2D}(\text{Reshape}([\mathbf{e_s}; \mathbf{r}]))$. Here, "Conv2D" is a 2D convolutional neural network (CNN) that operates over a representation of the subject and relation embeddings given by "Reshape". This latter operation first concatenates the two embeddings together before reshaping the result to be a square matrix for downstream convolutions. Furthermore, as in common in KGLP, we choose to train with the Binary Cross Entropy loss function, $\mathcal{L}_{\text{KGLP}} = \sum_{i=1}^{N} \text{BCE}(O_i, p(O_i|e_{s_i}, r_i))$.

4.2.2 Relation Extraction

In relation extraction, given a tokenized sentence $X = [x_1, \ldots, x_n], e_s, e_t$, the objective is to predict the relation, r, that best explains the connection between e_s and e_t in the sentence. Let $\mathbf{V} \in \mathbb{R}^{N_v \times d}$ denote learnable vocabulary embeddings corresponding to all words within a collection of documents with sentences, and $\mathbf{A}$ be a learned matrix of attributes (e.g. Named-Entity Recognition (NER) tags for each vocabulary word). We can then formulate a RE model as follows,

$$\mathbf{X} = \mathbf{V}X, \mathbf{A}X, \mathbf{s} = \mathbf{E}e_s, \mathbf{t} = \mathbf{E}e_t \qquad \texttt{Embedding Lookup} \qquad (4)$$

$$\hat{\mathbf{r}} = g(\mathbf{X}, \mathbf{A}, \mathbf{s}, \mathbf{t}) \qquad \texttt{Relation Prediction} \qquad (5)$$

$$p(r|\hat{\mathbf{r}}) = \text{Softmax}(\mathbf{R}\hat{\mathbf{r}} + \mathbf{b}) \qquad \texttt{Probability Estimation} \qquad (6)$$

Where $\hat{\mathbf{r}}$ is the inferred relation representation from a RE model g, and $p(r|\hat{\mathbf{r}})$ is the estimated probability distribution from g over all candidate relations. Note that in contrast to KGLP, where a variable amount of answers may be correct, in RE only a single relation may be correct. Figure 1 pictorially describes this framework.

PA-LSTM and RE Loss. The Position-Aware LSTM [20] (PA-LSTM) was originally proposed by [8] and formulates g as the combination of an LSTM network with a custom position-aware attention mechanism. Alongside the sentence tokens, it also utilizes NER, and positional embeddings describing the positional offset of each token from the respective subject and object respectively. Due to space limitations, we refer the readers to [8] for further information. Further, we use the canonical loss function used for training RE models: the Softmax Cross Entropy, $\mathcal{L}_{\text{RE}} = \sum_{i=1}^{N} = \text{SoftmaxCE}(r_i, p(r_i|\hat{\mathbf{r}_i}))$, where "SoftmaxCE" is the softmax cross entropy function and $p(r_i|\hat{\mathbf{r}_i})$ is as defined in 6.

4.2.3 Proposed Method: Merging KGLP and RE

Based on the observation that KGLP and RE are tightly coupled tasks, we explored a joint-learning framework that simultaneously optimized a KGLP and RE model with the aim of using the RE model to aid in the training of the KGLP method. Specifically, our framework trains each model in standard fashion using their respective objective functions, using a shared dictionary of embeddings and adding a third loss function that penalizes inconsistencies between predictions of the two models:

$$\mathcal{L}_{\text{COUPLING}} = \sum_{i=1}^{N} = \text{SoftmaxCE}(r_i, p_{\text{coupling}}(r_i|\hat{\mathbf{r}}_\mathbf{i})) \tag{7}$$

where $p_{\text{coupling}}(r_i|\hat{\mathbf{r}}_\mathbf{i}) = \text{Softmax}(\mathbf{R}\hat{\mathbf{r}}_\mathbf{i} + \mathbf{b}) = \text{Softmax}(\mathbf{R}g(\mathbf{X_i}, \mathbf{A_i}, \mathbf{s_i}, f(\mathbf{s_i}, \mathbf{r_i})) + \mathbf{b})$. In other words, in the coupling loss, we feed the output of a KGLP model (i.e. the predicted object representation) to an RE model that processes the expected object and subject in an associated sentence to estimate a relation encoding, which in turn should match the same relation used as input by the KGLP model. Figure 1 illustrates the method employed by this criterion.

Overall Objective Function. Our framework's overall objective function is formed by combining together the previously presented three criteria:

$$\mathcal{L} = \mathcal{L}_{\text{KGLP}} + \lambda_{\text{RE}}\mathcal{L}_{\text{RE}} + \lambda_{\text{COUPLING}}\mathcal{L}_{\text{COUPLING}} \tag{8}$$

Where $\lambda_{\text{RE}}, \lambda_{\text{COUPLING}} \geq 0$ are model hyperparmaters to be tuned. To reduce our hyperparameter search space, we set $\lambda_{\text{RE}} = \lambda_{\text{COUPLING}}$.

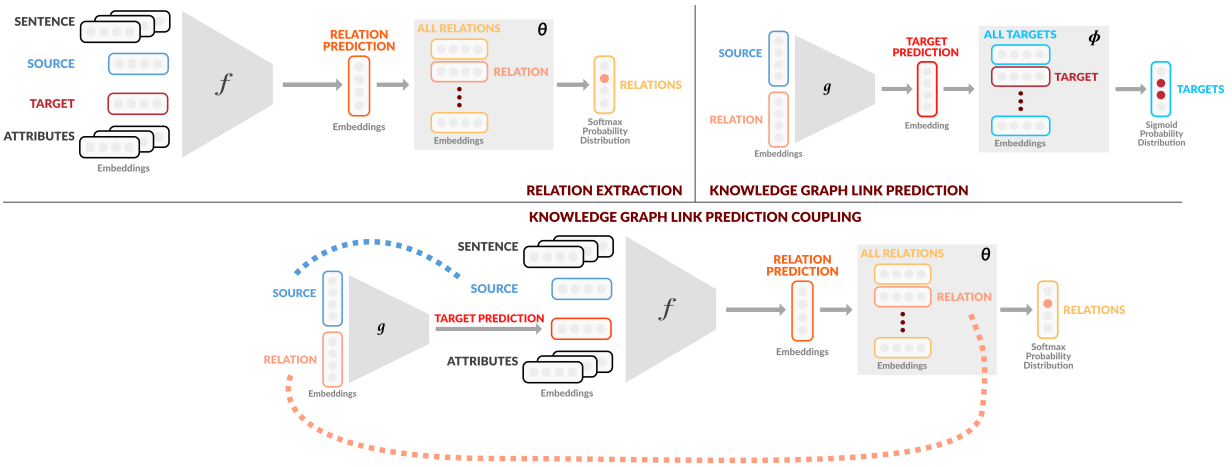

Figure 1: Top left: RE model framework. Top right: KGLP model framework. Bottom: Proposed joint model with shared entity and relation embeddings

5 Results

We present the results of the baseline KGLP model (ConvE) and our proposed joint KGLP + RE model. For simplicity, we fix λ_{RE} and λ_{KGLP} for our experiments. We evaluate the

performance on the KGLP task using the HITS@k (Recall@k) metric. Recall@k denotes the average number of times the correct target entity is among the top k ordered predictions given by our model. We report Recall@1 and Recall@10 for both the baseline ConvE model, as well as the proposed joint ConvE-PALSTM model we developed as part of this project. The table below summarizes the results

Model	λ_{RE}	$\lambda_{COUPLING}$	Recall@1	Recall@10
ConvE Baseline	0	0	8.11%	12.56%
ConvE-PALSTM	.1	.1	0.74%	5.98%
ConvE-PALSTM	1	1	16.73%	33.45%
Updated ConvE Baseline	0	0	11.56%	16.73%

Table 1: KGLP performance of ConvE baseline and proposed joint model

As shown in the table, we are able to achieve a Recall@1 performance of 8.11% on our baseline model compared to the reported performance of 23.7%. The discrepancy in performance arises due to the fact that we do not have access to the working code open sourced as part of the ConvE paper. Due to the packages becoming obsolete, our ability to run and reproduce the official code is severely limited. We made a few modifications to our implementation of ConvE and were able to obtain slightly improved performance. Due to lack of sufficient time, we were not able to run the ConvE-PALSTM model on the updated codebase. In this context, it is important to note that improvements in our baseline performance will directly translate to improvements in the proposed joint model. The improved baseline results are listed in the last row of 1

The key takeaways from our results in table 1 can be summarized as follows:

1. Our proposed joint model (ConvE-PALSTM) outperforms the ConvE baseline by a huge margin when a large enough λ_{RE} is chosen, validating our hypothesis that using a joint RE+KGLP model, we can improve the performance on the KGLP task. This suggests that a high value for λ_{RE} and $\lambda_{COUPLING}$ is essential to reap the benefits of the joint optimization task.

2. For smaller values of λ_{RE} and $\lambda_{COUPLING}$ the performance severely degrades, suggesting that the objectives of the RE and KGLP tasks are opposing each other due to insufficient weighting on the RE loss.

6 Conclusion

Starting with the observation that knowledge graph link prediction and relation extraction are tightly coupled, our project explored the joint optimization of both the tasks. Specifically, we proposed a joint-learning framework to simultaneously optimize the training of both KGLP and RE tasks with the aim of improving performance on KGLP using information from the RE task. Using a joint CNN-LSTM model, we notice a significant improvement on the KGLP task using the proposed method. Although our baseline performance does not match with the reported performance in literature, it is fair to assume that any improvements achieved on the baseline method will translate to improvements on the joint model.

References

[1] R. West, E. Gabrilovich, K. Murphy, S. Sun, R. Gupta, and D. Lin, "Knowledge base completion via search-based question answering," in *WWW*, 2014.

[2] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Advances in Neural Information Processing Systems*, pp. 2787–2795, 2013.

[3] B. Yang, W.-t. Yih, X. He, J. Gao, and L. Deng, "Embedding entities and relations for learning and inference in knowledge bases," in *International Conference on Learning Representations (ICLR)*, 2015.

[4] R. Wang, B. Li, S. Hu, W. Du, and M. Zhang, "Knowledge graph embedding via graph attenuated attention networks," *IEEE Access*, vol. 8, pp. 5212–5224, 2020.

[5] N. Lao, T. Mitchell, and W. W. Cohen, "Random walk inference and learning in a large scale knowledge base," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 529–539, Association for Computational Linguistics, 2011.

[6] M. Gardner, P. P. Talukdar, B. Kisiel, and T. Mitchell, "Improving learning and inference in a large knowledge-base using latent syntactic cues," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 833–838, 2013.

[7] P. Zhou, W. Shi, J. Tian, Z. Qi, B. Li, H. Hao, and B. Xu, "Attention-based bidirectional long short-term memory networks for relation classification," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, (Berlin, Germany), pp. 207–212, Association for Computational Linguistics, Aug. 2016.

[8] Y. Zhang, V. Zhong, D. Chen, G. Angeli, and C. D. Manning, "Position-aware attention and supervised data improve slot filling," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, (Copenhagen, Denmark), pp. 35–45, Association for Computational Linguistics, Sept. 2017.

[9] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, "Relation classification via convolutional deep neural network," in *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, (Dublin, Ireland), pp. 2335–2344, Dublin City University and Association for Computational Linguistics, Aug. 2014.

[10] M. Joshi, D. Chen, Y. Liu, D. S. Weld, L. Zettlemoyer, and O. Levy, "Spanbert: Improving pre-training by representing and predicting spans," *CoRR*, vol. abs/1907.10529, 2019.

[11] M. E. Peters, M. Neumann, R. L. L. IV, R. Schwartz, V. Joshi, S. Singh, and N. A. Smith, "Knowledge enhanced contextual word representations," 2019.

[12] J. Weston, A. Bordes, O. Yakhnenko, and N. Usunier, "Connecting language and knowledge bases with embedding models for relation extraction," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, (Seattle, Washington, USA), pp. 1366–1371, Association for Computational Linguistics, Oct. 2013.

[13] X. Han, Z. Liu, and M. Sun, "Neural knowledge acquisition via mutual attention between knowledge graph and text," in *AAAI*, 2018.

[14] G. Wang, W. Zhang, R. Wang, Y. Zhou, X. Chen, W. Zhang, H. Zhu, and H. Chen, "Label-free distant supervision for relation extraction via knowledge graph embedding," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, (Brussels, Belgium), pp. 2246–2255, Association for Computational Linguistics, Oct.-Nov. 2018.

[15] "Figure 1: Overview of JRRELP. JRRELP is comprised of three loss terms:...," Dec. 2020. [Online; accessed 4. May 2022].

[16] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: A collaboratively created graph database for structuring human knowledge," in *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, (New York, NY, USA), p. 1247–1250, Association for Computing Machinery, 2008.

[17] E. Sandhaus, "The new york times annotated corpus," *Linguistic Data Consortium, Philadelphia*, vol. 6, no. 12, p. e26752, 2008.

[18] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in *Association for Computational Linguistics (ACL) System Demonstrations*, pp. 55–60, 2014.

[19] T. Dettmers, M. Pasquale, S. Pontus, and S. Riedel, "Convolutional 2d knowledge graph embeddings," in *Proceedings of the 32th AAAI Conference on Artificial Intelligence*, pp. 1811–1818, February 2018.

[20] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.