Knowledge Graph Enhanced Relation Extraction

George Stoica
Carnegie Mellon University
5000 Forbes Ave,
Pittsburgh, PA 15213
`gis@cs.cmu.edu`

Emmanouil Antonios Platanios
Microsoft Semantic Machines
1 Microsoft Way,
Redmond, WA 98052
`emplata@microsoft.com`

Barnabás Póczos
Carnegie Mellon University
5000 Forbes Ave,
Pittsburgh, PA 15213
`bapoczos@cs.cmu.edu`

Abstract

Relation Extraction (RE) is the task of predicting a relation between a subject and object in a sentence, while knowledge graph link prediction (KGLP) infers a set of objects — O, given a subject and a relation from a knowledge graph. These two problems are closely intertwined: given a sentence consisting of a subject and object — o, a RE model estimates a relation that a KGLP model (together with the subject) may use to infer a set of objects — O — that contains o. In this paper, we leverage this insight by proposing a multi-task learning framework that enhances RE models by jointly training on both RE and KGLP tasks. We illustrate the generality of our approach by applying it on three existing RE methods and achieve consistent improvements across our benchmark datasets.

1 Introduction

Many real-world applications ranging from search engines to conversational agents rely on the ability to uncover new relationships from existing knowledge. Relation extraction (RE) and knowledge graph (KG) link prediction (KGLP) are two closely related tasks that center around inferring new information from existing facts. RE is the task of uncovering the relationship between two entities (termed the subject and object respectively) in a sentence. Similarly, KGLP involves inferring the set of correct answers (i.e., objects) to KG questions consisting of an entity (subject) and relation. These questions are given in triple-form: (`SUBJECT`, `RELATION`, ?). To illustrate their relationship, consider the sentence "`John and Jane are married`", whose subject and object are highlighted in blue and red respectively. Given this information, RE models infer the relationship between "`John`" and "`Jane`" (e.g., "`SPOUSE`"). Similarly, KGLP models infer the answers (objects) to the question (`John`, `SPOUSE`, ?). Based on the sentence, the answers must include "`Jane`". Thus, RE models predict the relation between a subject and object, while KGLP models infer the object from the subject and relation.

Several methods have been proposed to boost the performance of RE models by incorporating information from KGLP. However, these approaches typically require KGLP pre-training [32, 29], exhibit constrained parameter sharing [32, 29], or predominately attend over both problems through custom attention mechanisms [3, 11, 35]. Moreover, these frameworks only support a limited class of KGLP models that can be reframed as inferring relations from subject and objects. This constraint excludes recent KGLP methods which perform significantly better, but cannot be reformulated to satisfy the restriction. An ideal framework should support arbitrary RE and KGLP methods, including the significantly more expressive and stronger performing recent KGLP approaches. Additionally, such a framework should enable RE models to benefit from KGLP models with minimal changes to the underlying RE and KGLP methods.

We propose a general framework which ties the RE and KGLP tasks cohesively into a single learning problem. Our architecture, termed **JRRELP**—**J**ointly **R**easoning over **R**elation **E**xtraction with **L**ink **P**rediction—has the following desirable properties: (i) **General:** our method can be applied to arbitrary RE and KGLP models to boost RE performance, (ii) **Cyclical:** we enhance cross-task

4th Knowledge Representation and Reasoning Meets Machine Learning Workshop (KR2ML 2020), at NeurIPS.

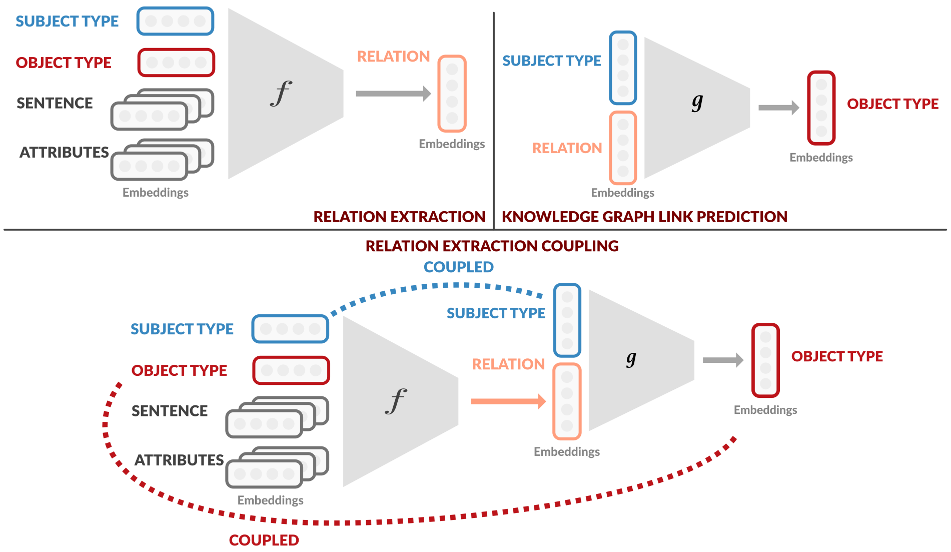

Figure 1: Overview of JRRELP. JRRELP is comprised of three tasks as described in Section 3: RE, KGLP and Coupling. The RE task is illustrated in the top-left quadrant, the KGLP task is described by the top-right quadrant, and the bottom half shows the Coupling task.

information sharing by cyclically coupling model parameters, (iii) **Scalable:** JRRELP introduces minimal overhead over baseline RE methods (only 6% slower batch updates). An overview of JRRELP is shown in Figure 1, and is explained in detail in Section 3.

2 Background

Let D represent a dataset composed of sentences $X = [x_1, x_2, \dots x_n]$, where x_j represents a one-hot encoding for the j^{th} sentence token (i.e., word). Each sentence contains a subject $s = [x_{s^{\text{start}}}, \dots, x_{s^{\text{end}}}]$, that is defined as a contiguous span $(s^{\text{start}}, s^{\text{end}})$ over the sentence, and an object $o = [o_{o^{\text{start}}}, \dots, o_{o^{\text{end}}}]$, that is similarly defined. Subjects and objects are defined by their *types*, termed s^{type} and o^{type}, respectively. Following our motivating example from Section 1, `John` and `Jane` would be tagged as having types `PERSON`. Several methods [36, 37, 9, 15] employ *type-substitution* during data preprocessing: substituting subjects and objects in sentences with their corresponding types. For instance, with type-substitution our example sentence becomes "`SUB:PERSON and OBJ:PERSON are married.`" Without loss of generality, we assume that sentences are preprocessed using type-substitution for the remainder of this paper. Finally, each sentence contains a relation, r, between its subject and object (e.g. `SPOUSE`).

Relation Extraction (RE). Given X, s and o, RE infers r between s and o. Many successful RE methods—including the current state-of-the-art [24]—involve learning vector embeddings for each component. Specifically, let N_v and N_r denote the vocabulary size for the sentence tokens and the number of unique relations respectively computed over a training dataset. Note that under our type-substitution assumption, N_v also contains all entity types. We define $\boldsymbol{V} \in \mathbb{R}^{D_v \times N_v}$ and $\boldsymbol{R} \in \mathbb{R}^{D_r \times N_r}$ as learnable vocabulary and relation embedding matrices respectively, where D_v and D_r denote the vocabulary and relation embedding sizes respectively. Given a sentence, a subject, an object, and a relation, their respective embedding representations are defined as: $\boldsymbol{X} = \boldsymbol{V}X \in \mathbb{R}^{D_v \times n}$, $\boldsymbol{s} = \boldsymbol{V}s \in \mathbb{R}^{D_v \times (s^{\text{end}} - s^{\text{start}} + 1)}$, $\boldsymbol{o} = \boldsymbol{V}o \in \mathbb{R}^{D_v \times (o^{\text{end}} - o^{\text{start}} + 1)}$, and $\boldsymbol{r} = \boldsymbol{R}r \in \mathbb{R}^{D_r}$, where n is the number of tokens in X. Given these embeddings, most successful RE models [36, 37, 9, 1, 24] can be formulated as instances of the following model,

$$\hat{\boldsymbol{r}} = f_{\text{RE}}(\boldsymbol{X}, \boldsymbol{s}, \boldsymbol{o}, \dots), \quad \text{and} \quad p_{\text{RE}}(r \mid \hat{\boldsymbol{r}}) = \text{Softmax}(\boldsymbol{R}\hat{\boldsymbol{r}} + \boldsymbol{b}), \tag{1}$$

where $\hat{\boldsymbol{r}}$ is the inferred relation representation from a prediction model f_{RE} and "..." accounts for any additional auxiliary information (attributes) that may be used (e.g. Part-of-Speech).

Knowledge Graph Link Prediction (KGLP). Given a question in the form of a subject-relation-object triple—$(s, r, ?)$, KGLP involves inferring the correct set of objects O. s and o are nodes in a KG, while r represents a graph edge between them. Although D does not explicitly specify a KG, one can be generated by extracting triples made up of sentence subjects, objects and relations. Specifically, given a sentence with s, o, and r, we can use the subject and object types—s^{type} and o^{type}, respectively—to form a KG whose edges are represented by r and nodes by s^{type} and o^{type}. For ease of notation, we assume that each term is a one-hot encoding of the corresponding identifier. Given this notation, we obtain KG component embeddings by: $\boldsymbol{s}^{\text{type}} = \boldsymbol{V} s^{\text{type}} \in \mathbb{R}^{D_v}$, $\boldsymbol{o}^{\text{type}} = \boldsymbol{V} o^{\text{type}}$, and $\boldsymbol{r} = \boldsymbol{R} r \in \mathbb{R}^{D_r}$. Multiple existing KGLP methods can be defined in terms of the following model:

$$\boldsymbol{z} = g_{\text{KGLP}}(\boldsymbol{s}^{\text{type}}, \boldsymbol{r}), \quad \text{and} \quad p_{\text{KGLP}}(O \mid o^{\text{type}}, \boldsymbol{z}) = \sigma(\boldsymbol{V}_{o^{\text{type}}} \boldsymbol{z} + \boldsymbol{b}) \tag{2}$$

where $\boldsymbol{z}$ is a merged representation of $\boldsymbol{s}^{\text{type}}$, g_{KGLP} is a KGLP model, and σ is the sigmoid activation function. While certain early KGLP methods [4, 33, 18, 14, 28] do not fit under this formulation, they may be accommodated by a simple reformulation of Equation 2 to their respective scoring terms.

3 Proposed Method

RE and KGLP tasks are tightly coupled. Given a sentence X with s and o, RE models predict the relationship—r—between s and o. Similarly, KGLP methods infer a set of objects O , where $o \in O$ (this is known because X describes this relationship) from s and r. JRRELP is a multi-task learning framework that explicitly accounts for the connection between RE and KGLP. JRRELP jointly trains a RE model, p_{RE}, and a KGLP model, p_{KGLP}, that are defined using our abstract formulation from Section 2 and optimized using one objective function. Below we describe each term of this function.

RE Loss. The first term corresponds to the standard loss function used to train RE models, $\mathcal{L}_{\text{RE}} = \sum_{i=1}^{N} \text{SCE}(r_i, p_{\text{RE}}(r_i \mid X_i, s_i, o_i, \ldots))$, where i denotes the i^{th} example in D, "SCE" represents the softmax cross-entropy loss function, and p_{RE} is defined as in Equation 1. Although this loss term assumes that a *single* relation exists between a subject and an object in a sentence, it is consistent with the loss term utilized by our baselines and is also appropriate for our widely used benchmark datasets described in Section 4. Additionally, we emphasize that this does not restrict the applicability of JRRELP to single-relation extraction problems. For instance, "SCE" can be substituted for binary-cross entropy (BCE) in the case of multi-label RE problems.

KGLP Loss. The second term corresponds to a popular loss function which is often used to train KGLP models. This loss function is defined as follows: $\mathcal{L}_{\text{KGLP}} = \sum_{i=1}^{N} \text{BCE}(O_i, p_{\text{KGLP}}(O_i \mid s_i^{\text{type}}, o_i^{\text{type}}, r_i))$, where p_{KGLP} is defined as in Equation 2. Note here that O_i is a set of objects that can be constructed automatically given all of the training data and conditioned on s_i^{type} and r_i, as described in Section 2. We also acknowledge that early KGLP methods [4, 33, 18, 14, 28] cannot be represented by this loss term. However, we emphasize that this does not detract from the generality of JRRELP because they can be integrated by changing this term to their respective objective functions.

Coupling Loss. The third term penalizes inconsistencies between the predictions of the RE and KGLP models, and is defined as follows: $\mathcal{L}_{\text{COUPLING}} = \sum_{i=1}^{N} \text{BCE}(O_i, p_{\text{COUPLING}}(O_i \mid X_i, s_i, o_i, s_i^{\text{type}}, o_i^{\text{type}}, \ldots))$, where $p_{\text{COUPLING}}(O_i \mid \ldots) = \sigma(\boldsymbol{V}_{o_i^{\text{type}}} g_{\text{KGLP}}(\boldsymbol{s}_{\boldsymbol{i}}^{\text{type}}, f_{\text{RE}}(\boldsymbol{X}_{\boldsymbol{i}}, \boldsymbol{s}_i, \boldsymbol{o}_i, \ldots)))$. The key difference between $\mathcal{L}_{\text{COUPLING}}$ and $\mathcal{L}_{\text{KGLP}}$ is that the relations embeddings, $\boldsymbol{r}_i$, computed by r_i in $\mathcal{L}_{\text{KGLP}}$, are replaced by the predicted relation embeddings $\hat{\mathbf{r}}_i$ from f_{RE}.

JRRELP Objective Function The JRRELP objective function is defined as follows:

$$\mathcal{L}_{\text{JRRELP}} = \mathcal{L}_{\text{RE}} + \lambda_{\text{KGLP}} \mathcal{L}_{\text{KGLP}} + \lambda_{\text{COUPLING}} \mathcal{L}_{\text{COUPLING}}, \tag{3}$$

where $\lambda_{\text{KGLP}} \geq 0$ and $\lambda_{\text{COUPLING}} \geq 0$ are model hyperparameters that may be tuned. Furthermore, we observed no negative impact in performance.

Most importantly, our framework introduces a cyclical relationship between the RE and KGLP models that couples them together very tightly. Specifically, the RE model predicts relation embeddings using $\boldsymbol{V}$ that it compares to $\boldsymbol{R}$ to produce distributions over relations. The KGLP model on the other hand predicts object embeddings using $\boldsymbol{R}$ that it compares to $\boldsymbol{V}$ to produce distributions over objects. It is mainly this cyclical relationship along with the coupling loss term that result in both the RE and KGLP models benefiting from each other and serves to enhance the performance and robustness of RE methods. An overview of JRRELP is shown in Figure 1.

Note that, even though JRRELP minimizes the joint three-task objective function shown in Equation 3, at test time we only use the RE model to predict relations between subjects and objects. Thus, JRRELP

Table 1: Results reported by our own experiments are marked by *. The remainder are taken from [1] and [23]. All numbers are expressed as percentages. † denotes experiments performed using additional data other than provided by the respective models. "–" denotes missing results from the respective publications. "SemEval-MM" denotes the Masked-Mention version of the SemEval dataset.

Dataset	**Metric**	**Models**								
		C-AGGCN	**TRE**	**BERT**$_{EM}$	**PA-LSTM**	$\boxed{\textbf{PA-LSTM}}$	**C-GCN**	$\boxed{\textbf{C-GCN}}$	**SpanBERT**	$\boxed{\textbf{SpanBERT}}$
TACRED	`Precision`	73.1	70.1	–	65.7	67.8*	69.9	**74.1***	69.2*	74.0*
	`Recall`	64.2	65.0	–	64.5	65.0*	63.3	61.9*	71.2*	67.3*
	`F1`	68.2	67.4	**71.5**†	65.1	66.4*	66.4	67.4*	70.2*	70.8*
SemEval-MM	`Precision`	–	–	–	75.2	74.8	76.5	76.9	81.2	82.7
	`Recall`	–	–	–	78.0	80.6	79.5	80.3	86.1	85.2
	`F1`	–	–	–	76.6	77.6	78.0	78.5	83.6	83.9

can be thought of as a framework which alters the learning trajectory of an RE model, rather than increase its capacity through using additional model parameters.

4 Experiments

Datasets. We empirically evaluate the performance of JRRELP over three existing relation extraction baselines on two widely used supervised benchmark datasets: TACRED [36] and SemEval 2010 Task 8 [12]. Consistent with prior literature [36, 37, 9], we report our metrics from the model with the median validation f1-score over five independent runs. Additionally, similar to [36, 37, 9, 24, 15, 1], we report our micro-averaged f1-scores on TACRED, and the macro-averaged scores on SemEval. Note that we evaluate using the *masked-mention* version of SemEval, which enforces type-substituted sentences. [37] showed this to be better suited to testing model generalizability. Our primary objective is to measure the importance of a joint RE and KGLP objective in environments where learning over both tasks is restricted *only* to data available in a relation extraction dataset. This helps us estimate how effective JRRELP may be in real-world applications where a pre-existing KG is not available.

Models. We illustrate the generality of JRRELP by evaluating it on baselines from both classes of RE approaches:[1] Two sequence-based models (PA-LSTM and SpanBERT), and a graph-based model (C-GCN). We join all three baselines with the KGLP method ConvE [6]. We distinguish between our baselines and their JRRELP variants by boxing their model names (e.g. $\boxed{\text{PA-LSTM}}$). Further details regarding integrating each model with JRRELP and hyperparameters can be found in Appendix A.

Results. We report our overall performance results on TACRED in Table 1. We observe that JRRELP consistently outperforms it's baseline variants over their `F1` and precision metrics. In particular, we find that JRRELP improves all baseline model performances by at least .6% `F1`, and yields improvements of up to 4.1% in precision. Moreover, to the best of our knowledge C-CGCN-JRRELP achieves a new state-of-the-art in precision. Furthermore, JRRELP bridges the performance gap between several methods, *without* altering their model capacities. Notably, $\boxed{\text{PA-LSTM}}$ matches the reported C-GCN performance, whose JRRELP variant matches TRE [1] — a significantly more expressive transformer-based approach. These results suggest that the true performance ceiling of reported relation extraction approaches may be significantly higher than their reported results, and that JRRELP serves as a conduit towards achieving these performances. Results on SemEval masked-mention indicate a similar pattern to TACRED: JRRELP improves performance across all baselines. This illustrates the effectiveness of JRRELP's framework in environments with little data.

5 Conclusion

We propose JRRELP, a novel framework that improves upon existing relation extraction approaches by leveraging insights from the complementary problem of knowledge graph link prediction. JRRELP bridges these two tasks through an abstract multi-task learning framework that jointly learns RE and KGLP problems by unconstrained parameter sharing. We exhibit this generality be extending three diverse relation extraction methods, and improve their performance. Specifically, JRRELP enhanced methods match or exceed more complex RE models, and achieve new state-of-the-art performance.

[1] Refer to Appendix C for their definitions.

Broader Impact

Relation extraction is pivotal to advancement of a diversity of applications ranging from improving artificial assistants and search engines, to medical functions such as automated drug and abnormal gene discovery. We propose an abstract framework that improves upon the performance of existing relation extraction (RE) methods by explicitly leveraging the similarities of relation extraction and knowledge graph (KG) link prediction, while only using the data provided in a RE dataset. Our modular approach is capable of enhancing many RE methods and requires minimal implementation changes, making it ideal for fast deployment across any of these applications to improve them. Moreover, by not being dependent on additional KG datasets, our framework becomes extremely flexible to many types of environments including unstructured text, which is very important in the medical domain. However, since our method can be applied to improve relation extraction from personal and sensitive data (e.g., determining a person's political affiliation), one needs to employ caution to ensure the privacy and protection of this data. This could also be avoided via proper government intervention and regulation of certain technologies and their uses.

References

[1] Christoph Alt, Marc Hübner, and Leonhard Hennig. Improving relation extraction by pre-trained language representations. *CoRR*, abs/1906.03088, 2019. URL `http://arxiv.org/abs/1906.03088`.

[2] Ivana Balazevic, Carl Allen, and Timothy Hospedales. TuckER: Tensor factorization for knowledge graph completion. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5185–5194, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1522. URL `https://www.aclweb.org/anthology/D19-1522`.

[3] Iz Beltagy, Kyle Lo, and Waleed Ammar. Improving distant supervision with maxpooled attention and sentence-level supervision. *CoRR*, abs/1810.12956, 2018. URL `http://arxiv.org/abs/1810.12956`.

[4] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, pages 2787–2795, 2013.

[5] Rajarshi Das, Shehzaad Dhuliawala, Manzil Zaheer, Luke Vilnis, Ishan Durugkar, Akshay Krishnamurthy, Alex Smola, and Andrew McCallum. Go for a walk and arrive at the answer: Reasoning over paths in knowledge bases using reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2018.

[6] Tim Dettmers, Minervini Pasquale, Stenetorp Pontus, and Sebastian Riedel. Convolutional 2d knowledge graph embeddings. In *Proceedings of the 32th AAAI Conference on Artificial Intelligence*, pages 1811–1818, February 2018. URL `https://arxiv.org/abs/1707.01476`.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL `http://arxiv.org/abs/1810.04805`.

[8] Matt Gardner, Partha Pratim Talukdar, Bryan Kisiel, and Tom 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*, pages 833–838, 2013.

[9] Zhijiang Guo, Yan Zhang, and Wei Lu. Attention guided graph convolutional networks for relation extraction. *CoRR*, abs/1906.07510, 2019. URL `http://arxiv.org/abs/1906.07510`.

[10] Kelvin Guu, John Miller, and Percy Liang. Traversing knowledge graphs in vector space. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 318–327, 2015.

[11] Xu Han, Zhiyuan Liu, and Maosong Sun. Neural knowledge acquisition via mutual attention between knowledge graph and text. In *AAAI*, 2018.

[12] Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/S10-1006`.

[13] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

[14] Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. Knowledge graph embedding via dynamic mapping matrix. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 687–696, 2015.

[15] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. Spanbert: Improving pre-training by representing and predicting spans. *CoRR*, abs/1907.10529, 2019. URL `http://arxiv.org/abs/1907.10529`.

[16] Ni Lao, Tom Mitchell, and William 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*, pages 529–539. Association for Computational Linguistics, 2011.

[17] Xi Victoria Lin, Richard Socher, and Caiming Xiong. Multi-hop knowledge graph reasoning with reward shaping. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3243–3253, 2018.

[18] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pages 2181–2187. AAAI Press, 2015. ISBN 0-262-51129-0. URL `http://dl.acm.org/citation.cfm?id=2886521.2886624`.

[19] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014. URL `http://www.aclweb.org/anthology/P/P14/P14-5010`.

[20] Arvind Neelakantan, Benjamin Roth, and Andrew McCallum. Compositional vector space models for knowledge base completion. In *ACL*, 2015.

[21] Thien Huu Nguyen and Ralph Grishman. Relation extraction: Perspective from convolutional neural networks. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 39–48, Denver, Colorado, June 2015. Association for Computational Linguistics. doi: 10.3115/v1/W15-1506. URL `https://www.aclweb.org/anthology/W15-1506`.

[22] Nanyun Peng, Hoifung Poon, Chris Quirk, Kristina Toutanova, and Wen tau Yih. Cross-sentence n-ary relation extraction with graph lstms, 2017.

[23] Matthew E. Peters, Mark Neumann, Robert L. Logan IV, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. Knowledge enhanced contextual word representations, 2019.

[24] Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. Matching the blanks: Distributional similarity for relation learning. *CoRR*, abs/1906.03158, 2019. URL `http://arxiv.org/abs/1906.03158`.

[25] George Stoica*, Otilia Stretcu*, Emmanouil Antonios Platanios*, Barnabás Póczos, and Tom M. Mitchell. Contextual Parameter Generation for Knowledge Graph Link Prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

[26] Kai Sheng Tai, Richard Socher, and Christopher D. Manning. Improved semantic representations from tree-structured long short-term memory networks. *CoRR*, abs/1503.00075, 2015. URL `http://arxiv.org/abs/1503.00075`.

[27] Kristina Toutanova, Victoria Lin, Wen-tau Yih, Hoifung Poon, and Chris Quirk. Compositional learning of embeddings for relation paths in knowledge base and text. In *ACL*, 2016.

[28] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *International Conference on Machine Learning (ICML)*, volume 48, pages 2071–2080, 2016.

[29] Guanying Wang, Wen Zhang, Ruoxu Wang, Yalin Zhou, Xi Chen, Wei Zhang, Hai Zhu, and Huajun 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*, pages 2246–2255, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1248. URL `https://www.aclweb.org/anthology/D18-1248`.

[30] Linlin Wang, Zhu Cao, Gerard de Melo, and Zhiyuan Liu. Relation classification via multi-level attention CNNs. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1298–1307, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1123. URL `https://www.aclweb.org/anthology/P16-1123`.

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

[32] Jason Weston, Antoine Bordes, Oksana Yakhnenko, and Nicolas 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*, pages 1366–1371, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/D13-1136`.

[33] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. In *International Conference on Learning Representations (ICLR)*, 2015.

[34] Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. Relation classification via convolutional deep neural network. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2335–2344, Dublin, Ireland, August 2014. Dublin City University and Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/C14-1220`.

[35] Ningyu Zhang, Shumin Deng, Zhanlin Sun, Guanying Wang, Xi Chen, Wei Zhang, and Huajun Chen. Long-tail relation extraction via knowledge graph embeddings and graph convolution networks. *CoRR*, abs/1903.01306, 2019. URL `http://arxiv.org/abs/1903.01306`.

[36] Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1004. URL `https://www.aclweb.org/anthology/D17-1004`.

[37] Yuhao Zhang, Peng Qi, and Christopher D. Manning. Graph convolution over pruned dependency trees improves relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2205–2215, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1244. URL `https://www.aclweb.org/anthology/D18-1244`.

[38] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo 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)*, pages 207–212, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-2034. URL `https://www.aclweb.org/anthology/P16-2034`.

Appendices

A Models Continued

A.1 Relation Extraction

PA-LSTM. This model was proposed by [36], and centers around formulating f_{RE} as the combination of a one-directional long short-term memory (LSTM) network, and a custom position-aware attention mechanism. In addition to the sentence, PALSTM also incorporates Part-of-Speech (POS) and Named-Entity-Reference (NER) tags, as well as tags representing the positional offset of each token from the subject and the object respectively. The method first applies the LSTM over the concatenated sentence, POS tag, and NER tag embeddings. A relation $\hat{\boldsymbol{r}}$ is then predicted by attending the LSTM outputs with a custom position-aware attention mechanism using the position offset tag embeddings.

C-GCN. This model was proposed by [37], and formulates f_{RE} as a graph-convolution network (GCN) over sentence dependency parse trees. It uses the same sentence attributes as PA-LSTM, and additionally the sentence dependency parse. Similar to PA-LSTM, the method first encodes a concatenation of the sentence, POS tag, and NER tag embeddings using a bi-directional LSTM network. The model then infers relations from these encodings by reasoning over the graph implied by a pruned version of the provided dependency tree parse. In particular, C-GCN computes the least common ancestor (LCA) between s and o, and uses positional offset tags to prune the tree around the LCA. Afterwards, C-GCN processes the sentence encodings using a graph convolution network (GCN) defined over the pruned dependency parse tree. The resulting representations are finally processed by a multi-layer perceptron to predict relations.

SpanBERT. This model was proposed by [15], and is one of the current state-of-the-art (SoTA) relation extraction methods. SpanBERT extends BERT [7] by pre-training at the span-level. Moreover, the model randomly masks contiguous text spans instead of individual tokens, and adds a span-boundary objective that infers masked spans from surrounding data. In contrast to PALSTM and C-GCN, SpanBERT only takes into account the sentence in its input to predict relations. Thus, f_{RE} is formulated as its complete architecture, without additional attributes. We chose this model because it is one of the current state-of-the-art RE models and it is also open-sourced, allowing to easily integrate it in our experimental evaluation pipeline.

Note that PA-LSTM, C-GCN, and SpanBERT are just three of many approaches supported by our abstract RE model formulation. For instance, other transformer-based methods [1, 24, 23] can also be represented by using a different definition for f_{RE}.

Hyperparameters. All model hyperparameters and training procedure can be found in our repository at `https://github.com/gstoica27/JRRELP.git`.

A.2 Knowledge Graph Link Prediction

ConvE. ConvE [6] is defined by using the following merge function in our abstract model formulation:

$$g_{\text{KGLP}}(\boldsymbol{s}^{\text{type}}, \boldsymbol{r}) = \text{Conv2D}(\text{Reshape}([s^{\text{type}}; \boldsymbol{r}]) \tag{4}$$

where "Conv2D" is a 2D convolution operation and "Reshape($[s^{\text{type}}; \boldsymbol{r}]$)" first concatenates s^{type} and $\boldsymbol{r}$ and then reshapes the resulting vector to be a square matrix, so that a convolution operation can be applied to it.

While we acknowledge that ConvE is not the current state-of-the-art (SoTA) KGLP approach, it performs very well while using only a fraction of the parameters current SoTA [25, 31] methods require, thus making it more efficient. Moreover, ConvE is an example of a KGLP method which cannot be restructured to infer r from s and o, making it infeasible to use with any of the previous joint RE and KGLP frameworks [29, 32]. Note that, our results can only be further enhanced by using a stronger KGLP approach and thus this choice should not affect our conclusions.

Table 2: TACRED and SemEval-MM F1 results from our ablation study. † denotes experiments conducted without $\mathcal{L}_{\text{COUPLING}}$, and ‡ marks those run without $\mathcal{L}_{\text{KGLP}}$.

Dataset	**Metric**	**Ablation Experiments**							
		PALSTM	**PA-LSTM**	**PALSTM** †	**PALSTM** ‡	**C-GCN**	**C-GCN**	**C-GCN** †	**C-CGCN** ‡
TACRED	F1	65.1	66.4	65.6	66.3	66.4	67.4	66.8	67.0
SemEval-MM	F1	76.6	77.6	76.8	77.3	78.0	78.5	78.1	78.4

B Experiments Continued

B.1 Ablation Experiments.

To examine the effects of JRRELP's $\mathcal{L}_{\text{KGLP}}$ and $\mathcal{L}_{\text{COUPLING}}$ over the traditional relation extraction objective, $\mathcal{L}_{\text{RE}}$, we perform an ablation study with each term removed on methods from both RE approach classes: sequence-based (PALSTM) and graph-based (C-GCN). Table 2 shows the F1 results. Metrics for each dataset are reported in the same manner as previous results. All ablation performances illustrate the importance of $\mathcal{L}_{\text{KGLP}}$ and $\mathcal{L}_{\text{COUPLING}}$ as part of JRRELP's framework, as their respective models are worse than the full JRRELP architecture; they exhibit performance drops up to .8% F1 respectively. Moreover, we observe the largest performance drop from the removal of $\mathcal{L}_{\text{COUPLING}}$ – which removes JRRELP's consistency constraint between RE and KGLP models. This highlights importance of establishing this relationship while training to achieve strong performance.

C Related Work

There are three areas of research that are related to the method we propose in this paper. In this section, we discuss related work in each area and position JRRELP appropriately.

Relation Extraction. Existing RE approaches can be classified in two categories: sequence-based, and graph-based methods. Given a sentence in the form of a sequence of tokens, sequence-based models infer relations by applying recurrent neural networks [38, 36], convolutional neural networks [34, 21, 30], or transformers [1, 24, 15, 23]. In addition to the sentence, graph-based methods use the structural characteristics of the sentence dependency tree to achieve strong performance. [22] apply an n-ary Tree-LSTM [26] over a split dependency tree, while [37, 9] employ a graph-convolution network (GCN) over the dependency tree.

Knowledge Graph Link Prediction. Existing KGLP approaches broadly fall under two model classes: single-hop and multi-hop. Given a subject and a relation, single-hop models infer a set of objects by mapping the subject and relation respectively to unique learnable finite dimensional vectors (embeddings) and jointly transforming them to produce an object set. These approaches can be translational [4] over the embeddings, multiplicative [33, 28], or a combination of the two [6, 18, 14, 2, 25, 31]. On the other hand, multi-hop approaches determine object sets by finding paths in the KG connecting subjects to the objects, and primarily consist of path-ranking methods [16, 8, 20, 10, 27, 5, 17].

Joint Frameworks. Several approaches [32, 11, 29, 35, 3] have explored using the additional supervision provided by a KG to benefit relation extraction model performance. Of these, we believe [32, 11, 29] are most similar to our work. [32] proposes a framework which utilizes a KGLP model, TransE [4], as an additional re-ranking term when evaluating an RE model. While employing TransE as a re-ranker improves performance, their framework trains TransE and the respective RE approach separately without parameter sharing. This only allows very restricted information sharing during evaluation. [11] proposes a dual-attention framework for jointly learning KGLP and RE tasks by computing a weight distribution over training data and shares parameters between tasks. However, like [32], [11] limits KGLP model selection to those which can reformulated as inferring relations from subjects and objects. This excludes a large number of recent methods [6, 2, 5, 17, 25, 31] which cannot be reframed in this way. [29] also presents a joint framework, LFDS, for training relation extraction approaches via KGLP objectives. In particular, the architecture introduces a similar objective to $\mathcal{L}_{\text{COUPLING}}$, but can only support the same class of KGLP methods as in [32, 11]. Moreover, LFDS requires KGLP pre-training, and does not share core parameters such as relation representations between RE and KGLP methods. This can create domain-shift between the two respective models and impact performance.

JRRELP improves upon previous literature by providing a single joint objective which simultaneously addresses all their aforementioned limitations. First, JRRELP proposes an abstract framework which supports many RE and KGLP methods through three standard-based loss terms. Second, JRRELP shares all its parameters between KGLP and RE tasks, and establishes a novel cyclical learning structure over core parameters. Third, RE and KGLP tasks are jointly trained without any problem-specific pretraining required, enabling tasks to benefit from each other simultaneously during training. Fourth, JRRELP's structure facilitates suport for RE and KGLP methods with minimal implementation changes: only requiring their respective substitutions into f and g.