

CARE: Co-Attention Network for Joint Entity and Relation Extraction

Wenjun Kong and Yamei Xia*

School of Computer Science, Beijing University of Posts and Telecommunications
{kongwenjun, ymxia}@bupt.edu.cn

Abstract

Joint entity and relation extraction is the fundamental task of information extraction, consisting of two subtasks: named entity recognition and relation extraction. However, most existing joint extraction methods suffer from issues of feature confusion or inadequate interaction between the two subtasks. Addressing these challenges, in this work, we propose a **Co-Attention** network for joint entity and **Relation Extraction** (CARE). Our approach includes adopting a parallel encoding strategy to learn separate representations for each subtask, aiming to avoid feature overlap or confusion. At the core of our approach is the co-attention module that captures two-way interaction between the two subtasks, allowing the model to leverage entity information for relation prediction and vice versa, thus promoting mutual enhancement. Through extensive experiments on three benchmark datasets for joint entity and relation extraction (NYT, WebNLG, and SciERC), we demonstrate that our proposed model outperforms existing baseline models. Our code will be available at <https://github.com/kwj0x7E/CARE>.

Keywords: Information extraction, Joint entity and relation extraction, Co-attention network

1. Introduction

Named entity recognition (NER) and relation extraction (RE) are fundamental for various NLP applications like information retrieval, sentiment analysis and question answering. These tasks empower computers to comprehend the underlying meanings within text, thereby enabling more sophisticated comprehension and manipulation of linguistic information. Figure 1 shows an example from the SciERC dataset (Luan et al., 2018), exemplifying the task of identifying named entities and entity-relation triplets.

Traditionally, NER and RE were approached as separate stages in a processing pipeline (Zhong and Chen, 2021). In this pipeline approach, the text would first undergo NER, where entities like names, locations, and dates are identified and categorized. Next, the output of NER would be passed to the relation extraction stage. The pipeline approach struggles to capture complex interactions between subtasks. This limitation arises due to the isolated nature of the pipeline stages, where errors or ambiguities from one stage can propagate to subsequent stages, leading to cascading errors.

End-to-end or joint modeling approaches (Eberts and Ulges, 2020; Wang and Lu, 2020; Wei et al., 2020; Yan et al., 2021; Wang et al., 2021) aim to mitigate these limitations. By learning the interdependencies between NER and RE tasks in a joint model, these approaches can better capture the intricate connections between entities and relations.

Unfortunately, existing joint approaches still exhibit certain limitations. Some methods (Bekoulis

Text		
Furthermore, we propose the use of standard parser evaluation methods for automatically evaluating the summarization quality of sentence condensation systems.		
Entity		Entity Type
parser evaluation methods		Method
summarization quality		Metric
sentence condensation systems		Method
Head Entity	Relation	Tail Entity
parser evaluation methods	Evaluate For	summarization quality
summarization quality	Evaluate For	sentence condensation systems

Figure 1: An example from the SciERC dataset (Luan et al., 2018). Entities are distinguished by being assigned various colors.

et al., 2018; Eberts and Ulges, 2020) encounter feature confusion due to the utilization of shared representations across both subtasks. In addition, most existing joint models (Bekoulis et al., 2018; Eberts and Ulges, 2020; Wei et al., 2020) fail to effectively model the interaction between NER and RE. Considering that entity information often carries significant relational indicators, and conversely, relation information contains valuable cues about entities, this oversight results in suboptimal performance.

To overcome the above limitations, effective approaches are needed that disentangle features, enhance representation learning, and foster a more harmonious interaction between the two interconnected tasks. To this end, we address the following two main challenges:

Firstly, how to alleviate feature confusion within the joint model when learning features across the two subtasks? Our approach involves employing task-specific multi-layer perceptrons, enabling

*Corresponding author.

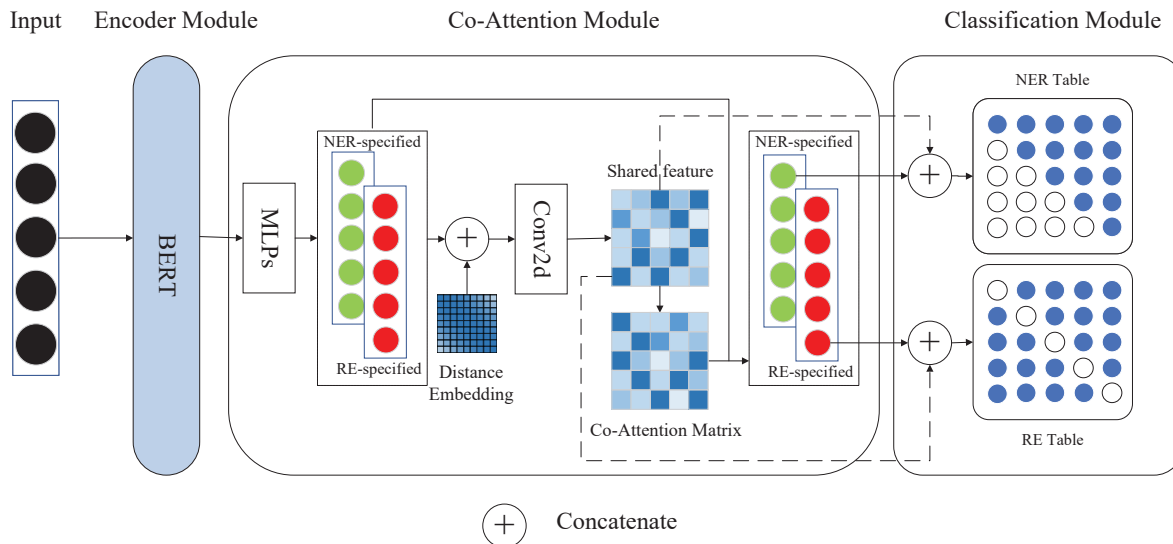


Figure 2: The overall framework of CARE, which consists of three modules: encoder module, co-attention module, and classification module. NER-specified and RE-specified representations are colored as green and red respectively.

the acquisition of task-specified representations for each subtask independently, thereby avoiding feature overlap or confusion.

Secondly, how to effectively model the intricate interaction between the two subtasks within a joint framework? We propose to employ a co-attention mechanism to model two-way interaction between NER and RE. Therefore, entity information can be harnessed to predict relations, and vice versa, enabling mutual enhancement between the two subtasks.

In summary, our main contributions are as follows:

- We propose CARE, a **Co-Attention** network for joint entity and **Relation** **E**xtraction, which can effectively exploit interactions between subtasks.
- Extensive experiments on three benchmark datasets show that our model can achieve superior performance compared with existing methods, and ablation studies demonstrate the effectiveness of our method.

2. Problem Definition

Joint Entity and Relation Extraction. The input of this problem is a sentence X consisting of N tokens. Let E denote a set of pre-defined entity types, and R denote a set of pre-defined relation types. Let $S = \{s_i\}_{i=1}^M$ be all the possible continuous spans in sentence X . We need to predict an entity type $e \in E$ for each span $s_i \in S$ in the sentence X . For spans of each predicted

entity pair $s_i, s_j \in S$, we need to predict relations $r \in R$. The output of this task is a set of entity-relation triplets $Y = \{(s_i, e_i), (s_j, e_j), r) : s_i, s_j \in S, e_i, e_j \in E, r \in R\}$.

3. Model

CARE consists of three modules: encoder module (Sec. 3.1), co-attention module (Sec. 3.2), classification module (Sec. 3.3), as is shown in Figure 2.

3.1. Encoder Module

Given a sentence X , the encoding module extracts contextual information from sentences. We use BERT (Devlin et al., 2019) as the pretrained encoder to produce contextual embeddings.

$$H = BERT(X) \quad (1)$$

3.2. Co-Attention Module

Typically, distinct feature and representation requirements emerge for NER and RE. The NER task may emphasize lexical-level features, whereas the RE task might prioritize contextual indications between entities. Relying solely on shared representations for these two subtasks can potentially lead to feature confusion.

Firstly, we adopt parallel encoding, using two distinct multi-layer perceptrons (MLPs) to acquire task-specific representations for individual subtasks,

thereby avoiding feature confusion.

$$\begin{aligned} H_{NER} &= MLP(H) \\ H_{RE} &= MLP(H) \end{aligned} \quad (2)$$

where H_{NER} represents NER-specified representations, and H_{RE} represents RE-specified representations.

To capture spatial semantic relationships among words, we introduce a matrix D for relative distance embeddings. Subsequently, we combine the task-specified representations with the relative distance embedding D , creating the input for the 3×3 convolutional layer. Inspired by Yan et al. (2021), we introduce the shared representation, which can convey valuable inter-task information to both subtasks. The feature map obtained from the 3×3 convolutional layer serves as the shared feature for both subtasks.

$$Q = [H_{NER}; H_{RE}; D] \quad (3)$$

$$H_{share} = Conv2d(Q) \quad (4)$$

For H_{share} , we use feed forward neural network (FFNN) and softmax activation function to calculate the co-attention score.

$$A = FFNN(H_{share}) \quad (5)$$

$$\alpha = Softmax(A) \quad (6)$$

$$\beta = Softmax(A^T) \quad (7)$$

where α represents entity-to-relation attention, and β represents relation-to-entity attention.

Next, we employ co-attention to enable interaction between task-specific representations. The results of the interaction are then integrated into the initial representations via skip connections, which are calculated as follows:

$$\begin{aligned} g_i^e &= h_i^e + \sum_{j=1}^n \alpha_{ij} h_j^r \\ g_i^r &= h_i^r + \sum_{j=1}^n \beta_{ij} h_j^e \end{aligned} \quad (8)$$

where $h_i^e, h_j^e \in H_{NER}$, $h_i^r, h_j^r \in H_{RE}$

By means of co-attention module, the task-specified representations H_{NER} and H_{RE} engage in a mutual interaction, thereby facilitating mutual enhancement between these two subtasks.

3.3. Classification Module

Following Yan et al. (2021), we formulate both NER and RE as table filling problems. This involves employing two-dimensional tables to represent entity labels and relation labels respectively. Note that NER can identify the entity span, the task of RE can be similarly regarded as a table filling problem,

where only the head word of the entity needs to be classified. We concatenate the task-specific representations with the shared representations, which serve as the input for the NER or RE classification module.

$$\begin{aligned} u_{ij}^e &= [g_i^e; g_j^e; h_{ij}^s] \\ u_{ij}^r &= [g_i^r; g_j^r; h_{ij}^s] \end{aligned} \quad (9)$$

where $h_{ij}^s \in H_{share}$.

The conditional probability $P(e|x_i, x_j)$ for the word pair (x_i, x_j) predicting the entity type as e , and $P(r|x_i, x_j)$ for the same word pair predicting the relation type as r are calculated as follows:

$$\begin{aligned} P(e|x_i, x_j) &= sigmoid(W_b \sigma(W_a u_{ij}^e + b_a) + b_b) \\ P(r|x_i, x_j) &= sigmoid(W_d \sigma(W_c u_{ij}^r + b_c) + b_d) \end{aligned} \quad (10)$$

where W and b are learnable parameters, $\sigma(\cdot)$ is activation function.

Finally, we regard both NER and RE as multi-label classification problems, and use binary cross-entropy as the loss of classification.

$$\begin{aligned} \mathcal{L}_{NER} &= - \sum_{i \leq j} \sum_{e \in E} \mathbb{I}(e = 1) \log P(e|x_i, x_j) \\ &\quad + \mathbb{I}(e = 0) \log(1 - P(e|x_i, x_j)) \\ \mathcal{L}_{RE} &= - \sum_{i \neq j} \sum_{r \in R} \mathbb{I}(r = 1) \log P(r|x_i, x_j) \\ &\quad + \mathbb{I}(r = 0) \log(1 - P(r|x_i, x_j)) \end{aligned} \quad (11)$$

where $\mathbb{I}(\cdot)$ denotes indication function.

The model is trained by the sum of \mathcal{L}_{NER} and \mathcal{L}_{RE} :

$$\mathcal{L} = \mathcal{L}_{NER} + \mathcal{L}_{RE} \quad (12)$$

4. Experiments

4.1. Setup

Datasets. We evaluate the performance of our model on three entity-relation extraction datasets NYT (Riedel et al., 2010), WebNLG (Gardent et al., 2017) and SciERC (Luan et al., 2018). The NYT dataset is a large-scale dataset for entity-relation extraction, which is collected from news articles from The New York Times and annotated by distant supervision, including 24 relation types. The WebNLG dataset was originally created for natural language generation tasks and was applied as a relation extraction dataset by Zeng et al. (2018), including 246 relation types. The SciERC dataset is collected from the abstracts of 500 artificial intelligence papers and used to build scientific knowledge graphs, including 6 entity types and 7 relation types. For all the datasets, we use the same split as the previous work (Yan et al., 2021).

Dataset	Method	PLM	NER	RE
NYT	CopyRL (Zeng et al., 2019)	-	-	72.1
	CasRel (Wei et al., 2020)	B	93.5	89.6
	TpLinker (Wang et al., 2020b)	B	-	91.9
	StereoRel (Tian et al., 2021)	B	-	92.2
	PFN (Yan et al., 2021)	B	95.8	92.4
	EmRel (Xu et al., 2022)	B	-	92.1
	CARE (Ours)	B	95.7	92.6
WebNLG	CopyRL (Zeng et al., 2019)	-	-	61.6
	CasRel (Wei et al., 2020)	B	95.5	91.8
	TpLinker (Wang et al., 2020b)	B	-	91.9
	StereoRel (Tian et al., 2021)	B	-	92.1
	PFN (Yan et al., 2021)	B	98.0	93.6
	EmRel (Xu et al., 2022)	B	-	92.9
	CARE (Ours)	B	98.1	93.9
SciERC	SPE (Wang et al., 2020a)	SciB	68.0	34.6
	UNIRE (Wang et al., 2021)	SciB	68.4	36.9
	PURE:single-sentence (Zhong and Chen, 2021)	SciB	66.6	35.6
	PURE:cross-sentence (Zhong and Chen, 2021)	SciB	68.9	36.8
	PFN (Yan et al., 2021)	SciB	66.8	38.4
CARE (Ours)	SciB	69.9	40.9	

Table 1: Performance on NYT, WebNLG and SciERC dataset. (PLM=pretrained language model, B=bert-base, SciB=scibert-base). The NER results of CasRel are reported from Yan et al. (2021).

Evaluation Metrics. Following the previous work (Yan et al., 2021), we evaluate the model using partial matching on the NYT and WebNLG datasets, where only the last word of an entity is annotated in the dataset. On the SciERC dataset, models are evaluated using strict matching, where the full span of an annotated entity is used. For evaluation metrics, we report Micro-F1 scores for NER and RE respectively. For the evaluation criteria for NER, an entity is considered correct only if its type and boundary are correct. For the evaluation criteria for RE, the predicted entity-relation triples are correct only when the entity types, entity boundaries and relation types are both correct. We run three times with different random seeds on each dataset and take the average as the result.

Implementation details. To ensure fair comparison with existing methods, we employ bert-base-cased (Devlin et al., 2019) for the NYT and WebNLG datasets, and scibert-scivocab-cased (Beltagy et al., 2019) for the SciERC dataset.

4.2. Main Results

Table 1 shows a comprehensive comparison between our model and existing methods. In both NER and RE tasks, when compared against the baseline model CasRel (Wei et al., 2020), CARE shows notable performance improvements. On the NYT dataset, CARE shows improvements of 2.2% and 3.0% for NER and RE respectively. On the WebNLG dataset, CARE outperforms CasRel

with gains of 2.6% for NER and 2.1% for RE. On the SciERC dataset, CARE also achieve superior performance, outperforming PFN (Yan et al., 2021) by 3.1% for NER and 2.5% for RE. What sets CARE apart from the majority of models is its capacity to capture the bidirectional interaction between entities and relations, achieved through the co-attention module. The aforementioned enhancements suggest that utilizing information from one subtask to augment the other yields significant benefits.

4.3. Ablation Study

We investigate the efficacy of various components within the model. These components include the relative distance embedding, co-attention mechanism, shared representation, 3×3 convolutional layers. The ablation results on the SciERC dataset are shown in Table 2. We find that: (1) Without the relative distance embeddings, there is a 0.6% decrease in NER and a 0.5% decrease in RE, which verified their useful in capturing the spatial relationships between words. (2) In the case of removing the shared representation from the classification layer, there is a decline of 0.5% in NER and 0.9% in RE, indicating the presence of interdependencies and correlations between subtasks. (3) When we replace the 3×3 convolutional layers with 1×1 convolutional layers, the performance of both NER and RE dropped. The 3×3 convolutional layers, with larger receptive field, are better suited to capture localized contextual information. (4) When we

Setting	NER	Δ	RE	Δ
Default	69.9	-	40.9	-
- Distance embedding	69.3	0.6	40.4	0.5
- Shared representation	69.4	0.5	40.0	0.9
- 3×3 convolution	69.1	0.8	40.2	0.7
- Co-attention	68.7	1.2	39.6	1.3

Table 2: Ablation study on the SciERC dataset.

Layers	NER	RE
N=1	69.5	40.3
N=2	69.6	40.6
N=3	69.9	40.9
N=4	68.2	40.5

Table 3: The impact of co-attention module depths on the SciERC dataset.

remove the co-attention module, a noticeable performance decline is observed. Specifically, there is a reduction of 1.2% in NER and 1.3% in RE, indicating that the co-attention module can effectively capture the two-way interaction between NER task and RE task.

Moreover, we investigate the influence of the co-attention module depth, as shown in Table 3. Notably, the best performance is achieved when the co-attention module comprises 3 layers. As the number of layers increases to 4, a reduction in performance becomes evident. It could potentially be attributed to the increased risk of overfitting as the number of layers grows, causing the loss of certain vital low-level features.

4.4. Case Study

We present a case study on the SciERC test set and make a comparison with the superior-performing model PFN (Yan et al., 2021) to demonstrate the advantages of CARE within specific scenarios. As shown in Figure 3, obviously, the PFN fails to recognize the entity "ray space" and the relation triplet ("ray space", "Feature-of", "geometric structures of 3D lines"), and the entity "ray space" is wrongly associated with another entity "geometric structures of 3D lines". In contrast, CARE successfully recognizes all entities and relations within the sentence, demonstrating a strong ability to harness the interaction between the two subtasks. Compared with the PFN, the utilization of relative distance embedding and 3×3 convolutions in our model enhances the capacity to capture interactions at the word-pair level, leading to improved accuracy in identifying entity boundaries.

5. Related Work

Most joint models can be classified into the following four categories: labeling-based, generation-

based, span-based and table filling.

Labeling-based. Bekoulis et al. (2018) adopt the BIO (Beginning, Inside, Outside) encoding scheme for NER, and regard RE as a multi-head selection problem. Wei et al. (2020) propose a cascaded binary labeling framework, wherein the head entity is initially extracted, followed by the extraction of the relation and its associated tail entity.

Generation-based. Zeng et al. (2018) propose a Seq2Seq relation extraction framework based on a copy mechanism. Zeng et al. (2019) apply reinforcement learning to a Seq2Seq model, considering the ordering of relation extraction. Cabot and Navigli (2021) view relation extraction as a Seq2Seq task, utilizing the pretrained model BART (Lewis et al., 2020) to generate relation triplets from the original sentence inputs.

Span-based. Luan et al. (2019) regard NER as a span classification problem and construct dynamic span graphs to produce span representations. Eberts and Ulges (2020) use max-pooling to produce span-level representations. Wang et al. (2020a) integrate span-level information into pretrained model for entity-relation extraction.

Table filling. Wang et al. (2020b) formulates joint extraction as a word-pair linking problem and introduced a handshake labeling scheme that aligns boundary words of entity pairs under each relation type. Wang and Lu (2020) designed two different encoders to capture table features and sequence features respectively. Wang et al. (2021) unify the label spaces of the two subtasks in a table. Yan et al. (2021) proposed a partition filtering network to model bidirectional interactions between subtasks.

Compared to PFN (Yan et al., 2021) and other approaches to entity-relation extraction, our proposed method features many different designs, each with unique motivations. First, the ways of learning task-specified features are different. We employ a parallel encoding approach, using two task-specific MLPs to extract task-specific features, whereas PFN only utilizes a single RNN-like encoder introducing entity and relation gates to filter out task-specific features. Specifically, the parallel encoding strategy enables direct learning of task-specific features for each subtask without extra filtering operations, thereby avoiding feature overlap. Second, the ways of modeling two-way interaction are different. In traditional parallel encoding approaches, task-specific features are independently generated using shared input. In order to model the interaction between the two subtasks based on parallel encoding, we introduces a co-attention

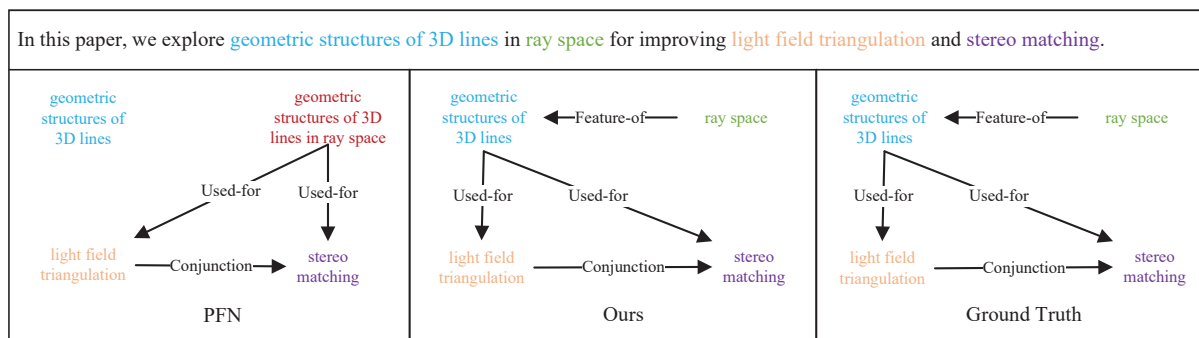


Figure 3: Case study on the SciERC test set. Entities are distinguished by being assigned various colors.

mechanism to selectively fuse all possible information from one subtask to enhance the other, thereby promoting mutual reinforcement.

6. Conclusion

In this paper, we propose CARE, a co-attention network designed for joint entity and relation extraction. We adopt parallel encoding to separately learn task-specific representations for NER and RE, preventing feature confusion between the two subtasks. Building upon parallel encoding, we introduce a co-attention mechanism to capture two-way interaction between NER and RE, effectively leveraging the information from one subtask to enhance the other. We evaluate our model through extensive experiments on three widely-used benchmark datasets (NYT, WebNLG and SciERC). The experimental results demonstrate that CARE achieves significant improvements over existing baseline methods.

7. Acknowledgements

We thank all the anonymous reviewers for their helpful comments and feedback. This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFF0902701, in part by the National Natural Science Foundation of China under Grants U21A20468, 62372058, U22A201339, 62162060, in part by the Fundamental Research Funds for the Central Universities under Grant 2020XD-A07-1.

8. Bibliographical References

- Giannis Bekoulis, Johannes Deleu, Thomas De-meester, and Chris Develder. 2018. Joint entity recognition and relation extraction as a multi-head selection problem. *Expert Systems with Applications*, 114:34–45.
- Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. 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 3615–3620.
- Pere-Lluís Huguet Cabot and Roberto Navigli. 2021. Rebel: Relation extraction by end-to-end language generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2370–2381.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.
- Markus Eberts and Adrian Ulges. 2020. Span-based joint entity and relation extraction with transformer pre-training. In *ECAI 2020*, pages 2006–2013. IOS Press.
- Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. Creating training corpora for nlg micro-planning. In *55th annual meeting of the Association for Computational Linguistics (ACL)*.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.
- Yi Luan, Luheng He, Mari Ostendorf, and Han-naneh Hajishirzi. 2018. Multi-task identification

- of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3219–3232.
- Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. 2019. A general framework for information extraction using dynamic span graphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3036–3046.
- Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2010, Barcelona, Spain, September 20-24, 2010, Proceedings, Part III 21*, pages 148–163. Springer.
- Xuetao Tian, Liping Jing, Lu He, and Feng Liu. 2021. Stereorel: Relational triple extraction from a stereoscopic perspective. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4851–4861.
- Jue Wang and Wei Lu. 2020. Two are better than one: Joint entity and relation extraction with table-sequence encoders. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1706–1721.
- Yijun Wang, Changzhi Sun, Yuanbin Wu, Junchi Yan, Peng Gao, and Guotong Xie. 2020a. Pre-training entity relation encoder with intra-span and inter-span information. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1692–1705.
- Yijun Wang, Changzhi Sun, Yuanbin Wu, Hao Zhou, Lei Li, and Junchi Yan. 2021. Unire: A unified label space for entity relation extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 220–231.
- Yucheng Wang, Bowen Yu, Yueyang Zhang, Tingwen Liu, Hongsong Zhu, and Limin Sun. 2020b. Tplinker: Single-stage joint extraction of entities and relations through token pair linking. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1572–1582.
- Zhepei Wei, Jianlin Su, Yue Wang, Yuan Tian, and Yi Chang. 2020. A novel cascade binary tagging framework for relational triple extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1476–1488.
- Benfeng Xu, Quan Wang, Yajuan Lyu, Yabing Shi, Yong Zhu, Jie Gao, and Zhendong Mao. 2022. Emrel: Joint representation of entities and embedded relations for multi-triple extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 659–665.
- Zhiheng Yan, Chong Zhang, Jinlan Fu, Qi Zhang, and Zhongyu Wei. 2021. A partition filter network for joint entity and relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 185–197.
- Xiangrong Zeng, Shizhu He, Daojian Zeng, Kang Liu, Shengping Liu, and Jun Zhao. 2019. Learning the extraction order of multiple relational facts in a sentence with reinforcement learning. 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 367–377.
- Xiangrong Zeng, Daojian Zeng, Shizhu He, Kang Liu, and Jun Zhao. 2018. Extracting relational facts by an end-to-end neural model with copy mechanism. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 506–514.
- Zexuan Zhong and Danqi Chen. 2021. A frustratingly easy approach for entity and relation extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 50–61.