

## Article

# Span-Based Fine-Grained Entity-Relation Extraction via Sub-Prompts Combination

Ning Yu <sup>1</sup>, Jianyi Liu <sup>2,\*</sup> and Yu Shi <sup>1</sup>

<sup>1</sup> Key Laboratory of Trustworthy Distributed Computing and Service (BUPT), Ministry of Education, Beijing University of Posts and Telecommunications, Beijing 100876, China

<sup>2</sup> School of Cyberspace Security (BUPT), Beijing University of Posts and Telecommunications, Beijing 100876, China

\* Correspondence: liujy@bupt.edu.cn

**Abstract:** With the development of information extraction technology, a variety of entity-relation extraction paradigms have been formed. However, approaches guided by these existing paradigms suffer from insufficient information fusion and too coarse extraction granularity, leading to difficulties extracting all triples in a sentence. Moreover, the joint entity-relation extraction model cannot easily adapt to the relation extraction task. Therefore, we need to design more fine-grained and flexible extraction methods. In this paper, we propose a new extraction paradigm based on existing paradigms. Then, based on it, we propose SSPC, a method for Span-based Fine-Grained Entity-Relation Extraction via Sub-Prompts Combination. SSPC first decomposes the task into three sub-tasks, namely  $\langle S, R \rangle$  Extraction,  $\langle R, O \rangle$  Extraction and  $\langle S, R, O \rangle$  Classification and then uses prompt tuning to fully integrate entity and relation information in each part. This fine-grained extraction framework makes the model easier to adapt to other similar tasks. We conduct experiments on joint entity-relation extraction and relation extraction, respectively. The experimental results show that our model outperforms previous methods and achieves state-of-the-art results on ADE, TACRED, and TACREV.

**Keywords:** joint entity and relation extraction; extraction paradigm; model adaptation; prompt tuning



**Citation:** Yu, N.; Liu, J.; Shi, Y.

Span-Based Fine-Grained Entity-Relation Extraction via Sub-Prompts Combination. *Appl. Sci.* **2023**, *13*, 1159. <https://doi.org/10.3390/app13021159>

Academic Editors: Gui-Lin Qi, Tong Xu and Meng Wang

Received: 5 December 2022

Revised: 10 January 2023

Accepted: 13 January 2023

Published: 15 January 2023



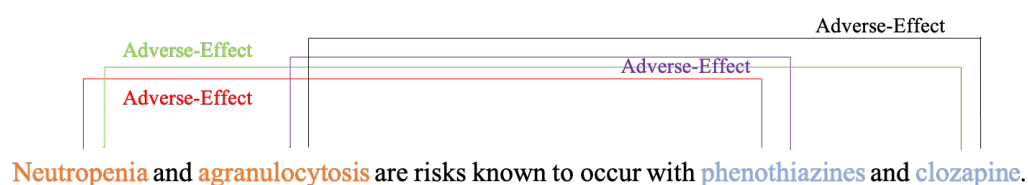
**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Information extraction is an important task in natural language processing, which aims to extract structured information from unstructured text. This task consists of two critical subtasks: Named Entity Recognition (NER) and Relation Extraction (RE). The former identifies various entities in a sentence, and the latter extracts whether and what relations exist between any two entities in a sentence. To extract complete structured information ( $\langle \text{Subject}, \text{Relation}, \text{Object} \rangle$ , i.e.,  $\langle S, R, O \rangle$ ), researchers usually handle these two complementary tasks simultaneously. So far, the resolution process has gone through three phases, evolving into three kinds of extraction paradigms. The two early paradigms can be formulated as “ $S, O \rightarrow [R]$ ” ([ ] represents a model) [1–3] and “[ $S, O$ ]  $\rightarrow [R]$ ” [4–7]. However, neither paradigm considers the connection between entities and relations, leading to poor extraction results. To address this problem, a lot of current work has focused on accomplishing both tasks in a single model, which can be called Entity and Relation Extraction (ERE). Thus, the third type of extraction paradigm evolved. Depending on what to extract first, it is further divided into the following specific forms, i.e.,  $P1: [S, O \rightarrow R]$ ,  $P2: [S \rightarrow R, O]$ , and  $P3: [R \rightarrow S, O]$  [8–15].

However, the third paradigm still has some drawbacks. Under the guidance of these existing paradigms, there are some problems in the extraction process, such as insufficient information fusion and coarse extraction granularity. Specifically, there is only a one-step information connection between entities and relations in the extraction process.  $P1$  and  $P2$  first extract the entities in the sentence and then transfer the information of the entities to

the relation extraction part. *P3* first extracts the relations in the sentence and then transfers the information of the relations to the entity extraction part. Nevertheless, in the first step of extraction, there is a lack of information supplement from relation information to entity extraction or entity information to relation extraction. Take the sentence “Neutropenia and agranulocytosis are risks known to occur with phenothiazines and clozapine.” in Figure 1 as an example. *P1* and *P2* are not guided by any relation information when extracting “Neutropenia”, “agranulocytosis”, “phenothiazines” and “clozapine” in the first step, even if the pre-defined target relation “Adverse-Effect” has been given in advance. The “Adverse-Effect” means “The effect was caused by the drug.”. The combination of “effect”, “drug”, and “was caused by” generates a large amount of information, including sequential information, causal information, and semantic information, which is very crucial for entity extraction. So, there is no reason to ignore this information, and we need to find a way to integrate it into the first step of the extraction process. Another issue is that the difference between relation extraction and joint entity and relation extraction is whether the entities in the sentence have been annotated. Therefore, a flexible ERE approach should ensure that it can quickly adapt to RE.



**Figure 1.** The example is chosen from the ADE dataset, where subjects are marked in orange, objects are marked in blue, and relations are marked with red, green, purple, and black lines.

To address the challenges above, we propose a new extraction paradigm, which can be described as  $[\langle S, R \rangle \& \langle R, O \rangle \rightarrow \langle S, R, O \rangle]$ . It can not only extract entities but also extract more fine-grained elements under the guidance of relation:  $\langle S, R \rangle$  and  $\langle R, O \rangle$ . As the intermediate extraction result, these elements can form triples through logical combination and specific judgment as the final joint extraction result. To implement it, we propose SSPC, a model for Span-based Fine-Grained Entity-Relation Extraction with Sub-Prompts Combination. We introduce the approach of prompt tuning into the model. A typical prompt consists of a template and a set of label words. This form ingeniously conforms to our idea of integrating entity and relation information as long as we convert the relation information into a prompt. Therefore, we first design a set of prompts before extraction, including the S-R prompt, R-O prompt, and S-R-O prompt, and then begin the extraction process. Firstly, traverse all spans of the sentence, and extract a set of possible subjects for each relation through the S-R prompt; Secondly, traverse all spans of the sentence, and extract a set of possible objects for each relation through the R-O prompt; Thirdly, combine the above intermediate results through a logic rule to form a set of undetermined triples, and filter each element in the set through the S-R-O prompt and relation classifier to identify the triples that indeed exist in the sentence. SSPC can be easily applied to the relation extraction task as well. Since the entities in the sentence are already annotated, it is not necessary to traverse all spans of the sentence but place the annotated entity in the corresponding position of the template. After the same steps, we can achieve relation extraction.

- We propose a new extraction paradigm. It solves the coarse-grained problem of the existing paradigm. A method guided by it can fully integrate entity and relation information in the extraction process to solve the problem of insufficient information fusion.
- We adopt the new extraction paradigm and propose SSPC, a model for Span-based Fine-Grained Entity-Relation Extraction with Sub-Prompts Combination. Moreover, this model can also be easily adapted to the relation extraction task. SSPC first

performs the  $\langle S, R \rangle$  extraction and then performs the  $\langle R, O \rangle$  extraction, and finally combines intermediate results through a logic rule to determine whether they are triples  $\langle S, R, O \rangle$  that are indeed contained in the sentence.

- We test our model on ADE, TACRED, and TACREV, and the results show that SSPC can significantly and consistently outperform existing state-of-the-art baselines.

## 2. Related Work

### 2.1. Joint Entity and Relation Extraction

The research direction of the joint entity and relation extraction is promising because it can solve the shortcomings of the previous methods, which ignore the interaction between sub-tasks and are affected by error propagation [16–20].

Under the guidance of the *P1* paradigm, Zeng et al. [8] proposed an end-to-end model based on sequence-to-sequence learning with copy mechanism, which can jointly extract relational facts from sentences of any of these classes. Both refs. [9,10] proposed a span-based joint entity and relation extraction model. The first work showed that with strong negative sampling, span filtering, and a localized context representation, a search over all spans in an input sentence becomes feasible for joint entity and relation extraction. The second work applied MLP attention to capture span-specific features aiming to obtain semantic rich span representation and calculated task-specific contextual representations with attention architecture to further reinforce span and relation representations. Unlike previous work based on BIO labels, these two approaches can identify overlapping entities and inspire us to think about the feasibility of the span-based approach.

Under the guidance of the *P2* paradigm, Li et al. [11] proposed a multi-turn question answering paradigm for the task of entity-relation extraction and Zhao et al. [12] provided an effective solution based on the machine reading comprehension models. These two models cast the entity-relation extraction to a multi-turn question answering (QA) and machine reading comprehension (MRC) task, enlightening us that joint extraction should adapt to more extraction paradigms and approaches.

Under the guidance of the *P3* paradigm, Chen et al. [13] proposed Patti, a novel pattern-first pipeline perspective for entity-relation extraction, which applied a Machine Reading Comprehension (MRC)-based framework to characterize entities and relations better. It alleviated entity redundancy and the entity overlap problem. Moreover, Takanobu et al. [14] applied a hierarchical reinforcement learning framework to enhance the interaction between entity mentions and relation types. Xie et al. [15] proposed a pipeline approach that first performs sentence classification with relational labels and then extracts the subjects and objects.

Recently proposed, the seq2seq model not only performs well in language generation but also in NLU tasks. Instead of tackling the joint extraction by training task-specific discriminative classifiers, Paolini et al. [21] framed it as a translation task between augmented natural languages, from which the task-relevant information can be easily extracted. Cabot et al. [22] performed relation extraction by representing triples as text sequences.

After analysis, under the guidance of existing paradigms, there is only a one-step information connection between entities and relations in the extraction process, resulting in insufficient information fusion. Moreover, these models are challenging to adapt to the relation extraction task.

### 2.2. Prompt Tuning

Recently, with the development of pre-trained language models, such as GPT [23], BERT [24], RoBERTa [25], T5 [26] and GPT-3 [27], the usage has also evolved from fine-tuning to prompt tuning. Prompt tuning has been widely used in various natural language processing tasks. Schick et al. [28] proposed an approach, PET, which consists of defining pairs of cloze question patterns and verbalizers that help leverage the knowledge contained within pre-trained language models for downstream tasks. It has achieved outstanding performance in widespread NLP tasks [29–31], especially in information extraction tasks [3,32–35]. Cui et al. [32] proposed a template-based method for NER, treating NER as

a language model ranking problem in a sequence-to-sequence framework, where original sentences and statement templates filled by candidate named entity span are regarded as the source sequence and the target sequence, respectively. Lee et al. [33] proposed demonstration-based learning, a simple-yet-effective way to incorporate automatically constructed auxiliary supervision. Instead of reformatting the NER task into the cloze-style template, they augment the original input instances by appending automatically created task demonstrations. Ma et al. [34] proposed a template-free prompt tuning method, EntLM, for few-shot NER. In this way, not only the complicated template-based methods can be discarded, but also the few-shot performance can be boosted since the model objective reduces the gap between pretraining and fine-tuning. These works demonstrate the diversity of prompt learning in information extraction tasks.

On the relation extraction task, refs. [3,35] give us the best inspiration. The first work proposed prompt tuning with rules for many-class text classification and applied logic rules to construct prompts with several sub-prompts. In this way, it can encode prior knowledge of each class into prompt tuning. Although this work is defined on the relation extraction task, we apply it to our framework through transformation. The second work incorporated knowledge among relation labels into prompt-tuning for relation extraction and proposes a knowledge-aware prompt-tuning approach with synergistic optimization. This approach is flexible. In the experiment, we also use this approach as a robust baseline.

### 3. Methodology

The overall framework of our approach is shown in Figure 2, in which different pre-trained language models (PLMs) are selected as the core according to the dataset characteristics.

Given a sentence:

$$Sent = \{w_1, w_2, w_3, \dots, w_n\},$$

we will focus on all spans in the sentence:

$$Span = \{w_i, w_{i+1}, w_{i+2}, \dots, w_{i+l}\},$$

where  $w$  denotes tokens,  $n$  denotes the sentence length and  $l$  denotes the span width. In this way, we will extract which span is the subject ( $S$ ) of a relation ( $R$ ) ( $\langle S, R \rangle$ ), which span is the object ( $O$ ) of a relation ( $R$ ) ( $\langle R, O \rangle$ ), and which two elements ( $\langle S, R \rangle$  and  $\langle R, O \rangle$ ) can form a triple ( $\langle S, R, O \rangle$ ).

For example, the relation “Adverse-Effect” in the ADE dataset means “The effect was caused by the drug.”. This explanation can guide joint entity and relation extraction and decompose the extraction process into:

1. If  $span_1$  is of type “Effect” and the sentence indicates that it was caused by a type of “Drug”, then  $span_1$  can be the subject of the relation:

$$\langle span_1, Adverse - Effect \rangle.$$

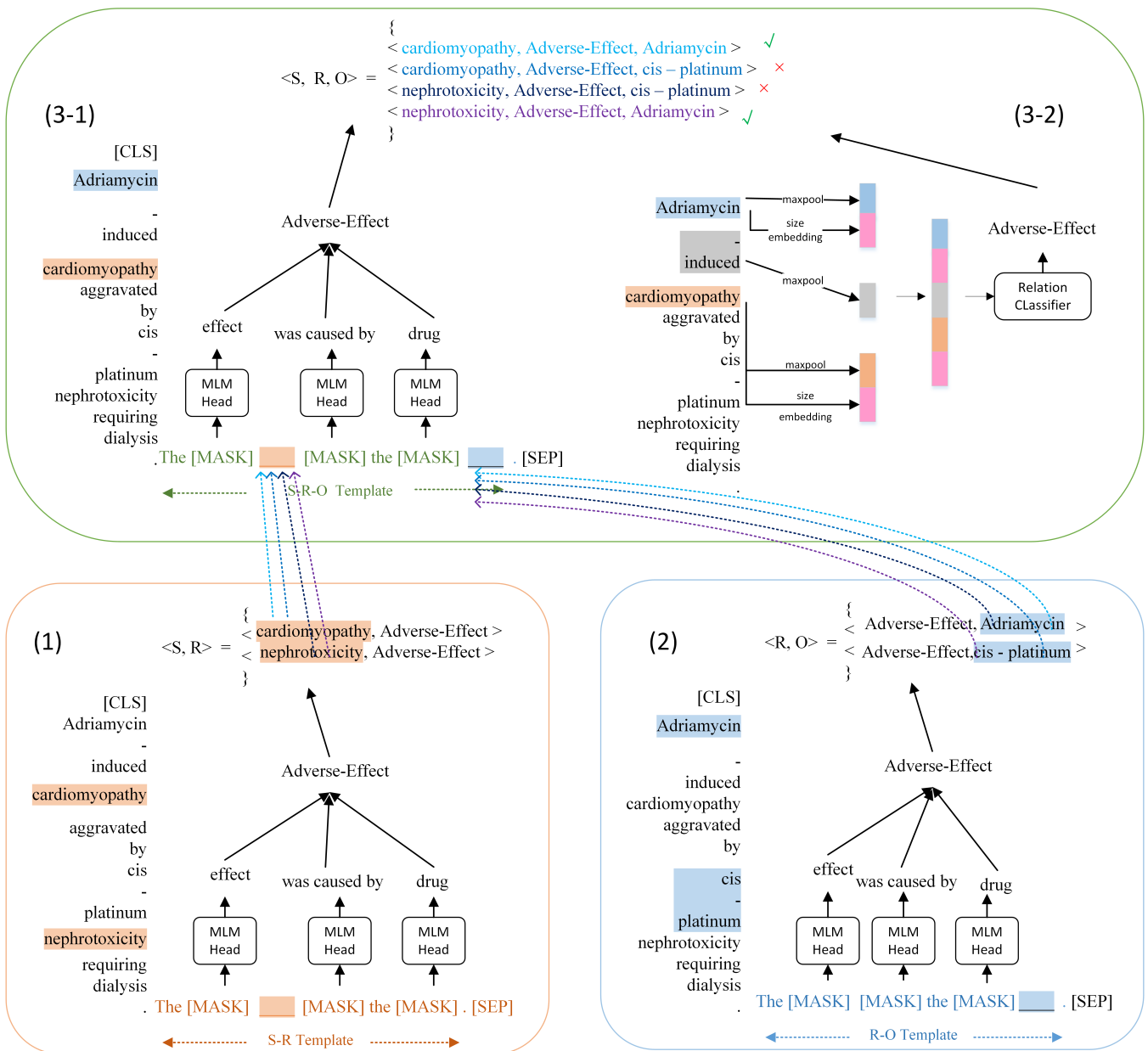
2. If  $span_2$  is of type “Drug” and the sentence indicates that it resulted in an “Effect”, then  $span_2$  can be the object of the relation:

$$\langle Adverse - Effect, span_2 \rangle.$$

3. If the sentence expresses that  $span_1$  and  $span_2$  have a “Cause” relation in the sentence, they can form a triple:

$$\langle span_1, Adverse - Effect, span_2 \rangle.$$

Based on the above considerations, we propose SSPC, an approach for Span-based Fine-Grained Entity-Relation Extraction with Sub-Prompts Combination. In the following sections, we describe each part of the framework in detail.



**Figure 2.** Our framework SSPC consists of three parts,  $\langle S, R \rangle$  Extraction (1),  $\langle R, O \rangle$  Extraction (2), and  $\langle S, R, O \rangle$  Classification (3-1) and (3-2).

### 3.1. Sub-Prompts Design

In order to achieve the extraction and classification tasks of the three parts and fully integrate the information of entities and relations, we naturally introduce the approach of prompt tuning. We design a set of prompts, including the S-R prompt, R-O prompt, and S-R-O prompt. Similar to the conventional prompt setting, each sub-prompt consists of a template and a set of label words.

We still take the ADE dataset as an example. Firstly, in preparation, we add the “no\_relation” relation, which means “The nothing is irrelevant to the nothing”. Then, the S-R Template, R-O Template, and S-R-O Template can be formalized as:

$$S-R \text{ Template}(span_1) = " \text{The } [MASK]_1 \text{ } span_1 [MASK]_2 \text{ the } [MASK]_3 \text{ . } " \quad (1)$$

$$R-O \text{ Template}(span_2) = " \text{The } [MASK]_1 [MASK]_2 \text{ the } [MASK]_3 \text{ } span_2 \text{ . } " \quad (2)$$

$$S\text{-}R\text{-}O \text{ Template}(span_1, span_2) = \text{“ The } [MASK]_1 \text{ } span_1 [MASK]_2 \text{ the } [MASK]_3 \text{ } span_2 \text{ .”} \quad (3)$$

The position of  $[MASK]_1$  can tell us the property of  $span_1$ ; the position of  $[MASK]_3$  can tell us the property of  $span_2$ ; the position of  $[MASK]_2$  can tell us the action of  $span_1$  or  $span_2$  and the interaction of  $span_1$  and  $span_2$  in the sentence. Therefore, to ensure grammatical correctness and semantic integrity, we design label words for  $[MASK]_1$ ,  $[MASK]_2$  and  $[MASK]_3$ , respectively, according to the characteristics of ADE dataset. The aggregated sets of label words are given as:

$$Labels\_ [MASK]_1 = \{ \text{“effect”, “nothing”} \} \quad (4)$$

$$Labels\_ [MASK]_2 = \{ \text{“was caused by”, “is irrelevant to”} \} \quad (5)$$

$$Labels\_ [MASK]_3 = \{ \text{“drug”, “nothing”} \} \quad (6)$$

Note that the set of label words of  $[MASK]_1$  is the same in the three templates, as are  $[MASK]_2$  and  $[MASK]_3$ . These three prompts will be applied to  $\langle S, R \rangle$  Extraction,  $\langle R, O \rangle$  Extraction, and  $\langle S, R, O \rangle$  Classification, respectively.

### 3.2. $\langle S, R \rangle$ Extraction

Given a sentence, we will traverse all spans of the sentence. We first put the current span  $\{w_i, w_{i+1}, w_{i+2}, \dots, w_{i+l}\}$  in the corresponding position of the S-R template and splice the sentence and the template to form

$$\{[CLS], \text{Sentence}, \text{Template}, [SEP]\},$$

and use the PLM to encode all tokens of the input sequence into responding vectors. Then, we take out the responding vectors  $\mathbf{h}_{[MASK]_1}$ ,  $\mathbf{h}_{[MASK]_2}$ , and  $\mathbf{h}_{[MASK]_3}$  of  $[MASK]_1$ ,  $[MASK]_2$ , and  $[MASK]_3$  and calculate the possibility of each label word in the corresponding labels set to fill in the position:

$$p([MASK] = w) = \frac{\exp(\mathbf{h}_{[MASK]} \cdot \mathbf{w})}{\sum_{\tilde{w} \in Labels\_ [MASK]} \mathbf{h}_{[MASK]} \cdot \tilde{\mathbf{w}}} \quad (7)$$

where  $\mathbf{w}$  is the embedding of the label word  $w$  in the PLM. Finally, we calculate the possibility of this span as the subject  $S$  of a relation  $\hat{R}$ :

$$\begin{aligned} p(\langle span, \hat{R} \rangle) &= p([MASK]_1 = labels\_ [MASK]_1[i]) \\ &\quad + p([MASK]_2 = labels\_ [MASK]_2[j]) \\ &\quad + p([MASK]_3 = labels\_ [MASK]_3[k]) \end{aligned} \quad (8)$$

where  $i$  is the index of the word of the  $\hat{R}$  at the  $[MASK]_1$  position in the  $Labels\_ [MASK]_1$ ;  $j$  is the index of the word of the  $\hat{R}$  at the  $[MASK]_2$  position in the  $Labels\_ [MASK]_2$ ;  $k$  is the index of the word of the  $\hat{R}$  at the  $[MASK]_3$  position in the  $Labels\_ [MASK]_3$ .

Thus, after traversing all spans, we will extract a set of possible subjects for each relation.

### 3.3. $\langle R, O \rangle$ Extraction

Similar to Section 3.2, we calculate the possibility of this span as the object  $O$  of a relation  $\hat{R}$ :

$$\begin{aligned} p(\langle \hat{R}, span \rangle) &= p([MASK]_1 = labels\_ [MASK]_1[i]) \\ &\quad + p([MASK]_2 = labels\_ [MASK]_2[j]) \\ &\quad + p([MASK]_3 = labels\_ [MASK]_3[k]) \end{aligned} \quad (9)$$

where  $i$  is the index of the word of the  $\hat{R}$  at the  $[MASK]_1$  position in the  $Labels_{-}[MASK]_1$ ;  $j$  is the index of the word of the  $\hat{R}$  at the  $[MASK]_2$  position in the  $Labels_{-}[MASK]_2$ ;  $k$  is the index of the word of the  $\hat{R}$  at the  $[MASK]_3$  position in the  $Labels_{-}[MASK]_3$ .

Thus, after traversing all spans, we will extract a set of possible objects for each relation.

### 3.4. $\langle S, R, O \rangle$ Classification

After calculation and reasoning in Sections 3.2 and 3.3, we obtain all the  $\langle S, \hat{R} \rangle$  elements and  $\langle \hat{R}, O \rangle$  elements in the sentence as intermediate results. We apply a simple strategy of using a logic rule and directly concatenate  $\langle S, \hat{R} \rangle$  and  $\langle \hat{R}, O \rangle$  of the same  $\hat{R}$  to form  $\langle S, \hat{R}, O \rangle$  as the element in the set of pending triples. Then, we filter each element through the S-R-O prompt and relation classifier to identify the triples that really exist in the sentence.

On the one hand, we put  $S$  and  $O$  in the positions of  $span_1$  and  $span_2$  in the S-R-O template. Similar to Sections 3.2 and 3.3, we calculate the possibility of  $S$  and  $O$  as the subject and object of a relation  $\hat{R}$ , respectively:

$$p(\langle span_1, \hat{R}, span_2 \rangle) = p([MASK]_1 = labels_{-}[MASK]_1[i]) + p([MASK]_2 = labels_{-}[MASK]_2[j]) + p([MASK]_3 = labels_{-}[MASK]_3[k]) \tag{10}$$

where  $i$  is the index of the word of the  $\hat{R}$  at the  $[MASK]_1$  position in the  $Labels_{-}[MASK]_1$ ;  $j$  is the index of the word of the  $\hat{R}$  at the  $[MASK]_2$  position in the  $Labels_{-}[MASK]_2$ ;  $k$  is the index of the word of the  $\hat{R}$  at the  $[MASK]_3$  position in the  $Labels_{-}[MASK]_3$ .

Thus, we will give the first relation prediction  $\mathbf{R}_{pred_1}$  about this entity pair  $span_1$  and  $span_2$ .

On the other hand, we fuse the information of  $S$  and  $O$  and input them into the relation classifier to predict whether they have a relation and what kind of relation they have. The input consists of five parts:

1. The embedding of  $span_1$ . All the token embeddings are combined using a fusion,  $f(e_i, e_{i+1}, e_{i+2}, \dots, e_{i+l})$ . Regarding the fusion function  $f$ , we choose max-pooling, obtaining the  $span_1$ 's representation  $\mathbf{e}(span_1)$ .
2. The size embedding of  $span_1$ . Given the span size  $l$ , we look-up a size embedding from a dedicated embedding matrix, obtaining the  $span_1$ 's size representation  $\mathbf{s}(span_1 \text{ size})$ . These size embeddings are learned by backpropagation.
3. The embedding of context. Obviously, words from the context are essential indicators of the expressed relation. We use a more localized context drawn from the direct surrounding of the spans: Given the span ranging from the end of the  $span_1$  to the beginning of the  $span_2$ , we combine its embeddings by max-pooling, obtaining a context representation  $\mathbf{c}(span_1, span_2)$ . If the range is empty (e.g., in case of overlapping entities), we set  $\mathbf{c}(span_1, span_2) = \mathbf{0}$ .
4. The embedding of  $span_2$ . Similar to the  $span_1$ 's embedding, we obtain the  $span_2$ 's representation  $\mathbf{e}(span_2)$ .
5. The size embedding of  $span_2$ . Similar to the  $span_1$ 's size embedding, we obtain the  $span_2$ 's size representation  $\mathbf{s}(span_2 \text{ size})$ .

The final input to the relation classifier is (whereas  $\circ$  denotes concatenation):

$$X_{(span_1, span_2)} = \mathbf{e}(span_1) \circ \mathbf{s}(span_1 \text{ size}) \circ \mathbf{c}(span_1, span_2) \circ \mathbf{e}(span_2) \circ \mathbf{s}(span_2 \text{ size}) \tag{11}$$

Then, it was passed through a single-layer classifier:

$$\hat{y}_{(span_1, span_2)} = \sigma(w \cdot X_{(span_1, span_2)} + b) \tag{12}$$

where  $\sigma$  denotes a softmax function or a sigmoid function. The highest response in the layer indicates that the corresponding relation  $\mathbf{R}_{pred_2}$  holds between  $span_1$  and  $span_2$ .

Finally, if  $\hat{R}$ ,  $R_{pred_1}$ , and  $R_{pred_2}$  are equal, we will choose this  $\langle S, \hat{R}, O \rangle$  as one of the results of the entity relation extraction of the sentence.

Thus, after traversing all pending triples, we will obtain a set of actual triples in the sentence, which completes the entire entity relation extraction process.

### 3.5. Joint Training

Our proposed framework consists of three parts, which are trained jointly. We define the loss function of the overall framework:

$$L = L_{\langle S, R \rangle} + L_{\langle R, O \rangle} + L_{\langle S, R, O \rangle} \quad (13)$$

The first term is defined on the  $\langle S, R \rangle$  Extraction. It includes two parts, one is the sum of cross-entropy loss of each masked position label, the other is the cross-entropy loss of span label. The second term is defined on the  $\langle R, O \rangle$  Extraction, the loss is the same as the first term. The third term is defined on the  $\langle S, R, O \rangle$  Extraction. It includes two parts, one is the same as the first term, the other is the cross-entropy loss of the relation label of  $span_1$  and  $span_2$ .

For all these three parts, we construct many within-sentence negative examples to train:

- For the  $\langle S, R \rangle$  Extraction and  $\langle R, O \rangle$  Extraction, We construct  $n$  within-sentence negative examples (“no\_relation”) by
  1. replacing the subject and object of the R (the subject acts as the object, and the object acts as the subject),
  2. blurring the subject or object’s boundary of the R (expanding or narrowing the starting and ending positions of a span), and
  3. generating another unrelated span as the subject or the object of the R.

These negative examples are combined with the existing positive examples to form a training set.

- For the  $\langle S, R, O \rangle$  Extraction, We construct  $n$  within-sentence negative examples (“no\_relation”) by
  1. replacing the subject and object of a single triple,
  2. cross replacing the subjects and objects of multiple triples,
  3. blurring the subject or object’s boundary of a single triple, and
  4. generating other unrelated spans as subjects and objects to form triples.

These negative examples are combined with the existing positive examples to form a training set.

- For example, given the sentence, “Adriamycin—induced cardiomyopathy aggravated by cis—platinum nephrotoxicity requiring dialysis.”, we will construct negative samples

$$\langle \text{Adriamycin}, \text{no\_relation} \rangle, \langle \text{platinum nephrotoxicity}, \text{no\_relation} \rangle, \\ \langle \text{aggravated}, \text{no\_relation} \rangle, \text{etc.}$$

for the  $\langle S, R \rangle$  Extraction;

$$\langle \text{no\_relation}, \text{cardiomyopathy} \rangle, \langle \text{no\_relation}, \text{platinum} \rangle, \\ \langle \text{no\_relation}, \text{dialysis} \rangle, \text{etc.}$$

for the  $\langle R, O \rangle$  Extraction;

$$\langle \text{Adriamycin}, \text{no\_relation}, \text{cardiomyopathy} \rangle, \\ \langle \text{cardiomyopathy}, \text{no\_relation}, \text{cis} - - \text{platinum} \rangle, \\ \langle \text{nephrotoxicity}, \text{no\_relation}, \text{platinum} \rangle, \langle \text{induced}, \text{no\_relation}, \text{dialysis} \rangle, \text{etc.}$$

for the  $\langle S, R, O \rangle$  Extraction.



### 3.6. Adapt to the Relation Extraction Task

The difference between relation extraction and joint entity relation extraction is whether entities in the sentence have been annotated. Therefore, a flexible ERE approach should ensure that it can quickly adapt to the RE. Our proposed SSPC guarantees this. Since the entities in the sentence are already annotated, it is not necessary to traverse all spans of the sentence but place the annotated entity in the corresponding position of the template. After the same steps, we can achieve relation extraction.

## 4. Experiments

### 4.1. Datasets

Our experiments are conducted on three widely used benchmarks: ADE, TACRED and TACREV.

- **ADE:** The ADE dataset [36] is about drugs and their adverse effects. A sentence contains entities of drug type and adverse effect type and expresses the corresponding relation between them. It consists of 4,272 sentences, of which 1,695 contain overlapping phenomena. As shown in Table 1, there are triples in the ADE dataset. It can be seen that even if only one relation is involved, entities and relations have become complex, mainly including multiple triples (1), subjects and  $\langle R, O \rangle$  many-to-one (2),  $\langle S, R \rangle$  and objects one-to-many (3), and subjects and objects crossing (4). As in previous work, we conduct 10-fold cross-validation. The core pre-trained language model is “microsoft/BiomedNLP-PubMedBERT-base-uncased-abstract-fulltext”; the maximum length of a sentence is set to 128; the maximum length of a span is set to 5; the number of within-sentence negative samples is set to 70. The experiments are conducted on an NVIDIA Geforce RTX 3090 24GB GPU. Other hyperparameters are shown in Table 2.
- **TACRED and TACREV:** The TACRED dataset [37] is a sizeable relation extraction dataset with 106,264 examples (68,124 for training, 22,631 for validation, and 15,509 for testing). These examples are created by combining available human annotations from the TAC KBP challenges and crowdsourcing. It contains 42 relation types, such as “per:parents”, “org:website”, and “no\_relation”. The TACREV dataset [38] is built based on the original TACRED dataset. It finds out and corrects the errors in the original development set and test set of TACRED, while the training set is left intact. To experiment, we set entity types according to relation types, including “person”, “organization”, “religion”, “country”, “state”, “city”, “title”, “number”, “URL”, “event”, and “date”. For few-shot learning, consistent with the previous work, we sample  $K$  training and  $K$  validation instances per class from the original training and development set and evaluate models on the original test set. We set  $K$  from {8, 16, 32}, respectively. The core pre-trained language model is “roberta-large”; the maximum length of a sentence is set to 256; the maximum length of a span is set to 5. The experiments are conducted on an NVIDIA Geforce RTX 3090 24GB GPU. Other hyperparameters are shown in Table 2.

**Table 1.** Triples in the ADE dataset, including multiple triples (1), subjects and (R, O) many-to-one (2), (S, R) and objects one-to-many (3), and subjects and objects crossing (4). The subjects are marked in orange, and the objects are marked in blue.

1	Adriamycin—induced cardiomyopathy aggravated by cis—platinum nephrotoxicity requiring dialysis . { <cardiomyopathy, Adverse-Effect, Adriamycin>, <nephrotoxicity, Adverse-Effect, cis—platinum> }	
2	Possible interaction between lopinavir / ritonavir and valproic Acid exacerbates bipolar disorder . { <bipolar disorder, Adverse-Effect, lopinavir>, <bipolar disorder, Adverse-Effect, ritonavir>, <bipolar disorder, Adverse-Effect, valproic Acid> }	
3	Listeria brain abscess , Pneumocystis pneumonia and Kaposi 's sarcoma after temozolomide . { <Listeria brain abscess, Adverse-Effect, temozolomide>, <Pneumocystis pneumonia, Adverse-Effect, temozolomide>, <Kaposi 's sarcoma, Adverse-Effect, temozolomide> }	
4	Neutropenia and agranulocytosis are risks known to occur with phenothiazines and clozapine . { <Neutropenia, Adverse-Effect, phenothiazines>, <Neutropenia, Adverse-Effect, clozapine>, <agranulocytosis, Adverse-Effect, phenothiazines>, <agranulocytosis, Adverse-Effect, clozapine> }	

**Table 2.** Hyperparameters for the different datasets. (Setting:few-shot = †)

	Max epochs	Learning rate	Warm-up	Weight decay	Batch size	Time per epoch
ADE	20	$5 \times 10^{-5}$	10%	0.01	1	46 min $\times$ 10
TACRED	20	$3 \times 10^{-5}$	10%	0.01	8	102 min
TACREV	20	$3 \times 10^{-5}$	10%	0.01	8	102 min
TACRED †	50	$3 \times 10^{-5}$	10%	0.01	8	0.5–2 min
TACREV †	50	$3 \times 10^{-5}$	10%	0.01	8	0.5–2 min

#### 4.2. Comparison with the State-of-the-Art Model of the ADE Dataset

In this part, we compare SSPC with other joint entity relation extraction models, including the state-of-the-art model:

- Natural Language Understanding (NLU) models**, including “CNN + Global features” proposed by [16], “BiLSTM + SDP” proposed by [17], “Multi-head” proposed by [18], “Multi-head + AT” proposed by [19], “Relation-Metric” proposed by [20], “SpERT” proposed by [10], “SPAN” proposed by [9]. “SpERT” and “SPAN” are both span-based joint entity and relation extraction models. “SpERT” is a span-based joint entity and relation extraction with transformer pre-training model; “SPAN” is a span-based joint entity and relation extraction with attention-based span-specific and contextual semantic representations model and is the current span-based state-of-the-art model on the ADE dataset.
- Natural Language Generation (NLG) models**, including “TANL” proposed by [21] and “REBEL” proposed by [22]. Recently, the seq2seq model not only performs well in language generation but also in NLU tasks. “TANL” is generative as it translates from an input to an output in augmented natural languages. “REBEL” is a seq2seq model based on BART that performs end-to-end relation extraction and is the current state-of-the-art model on the ADE dataset.

Table 3 shows our experimental results. The second column is the names of the baseline models. On the ADE dataset, we use standard *Precision*, *Recall*, and *F1* for evaluation. Furthermore, when *S* and *O*'s boundaries are correct, we treat a  $\langle S, R, O \rangle$  as correct. We

report the mean performance over 10-fold cross-validation to compare results to established work. From the results, we can see that SSPC consistently outperforms both NLU and NLG SOTA models. Compared with SPAN in entity recognition, *Precision* increases by 0.78% and *F1* increases by 0.4%. Compared with SPAN in relation extraction, *Precision* increases by 1.23%, *Recall* increases by 0.93%, and *F1* increases by 1.06%. Compared with REBEL in relation extraction, *F1* increases by 0.1%. We owe these performance increases to our new extraction paradigm and fine-grained extraction framework.

**Table 3.** Main experimental results on the ADE dataset. The best results are bold.

Dataset	Model	Entity			Relation		
		Precision	Recall	F1	Precision	Recall	F1
ADE	CNN + Global features [16]	79.50	79.60	79.50	64.00	62.90	63.40
	BiLSTM + SDP [17]	82.70	86.70	84.60	67.50	75.80	71.40
	Multi-head [18]	84.72	88.16	86.40	72.10	77.24	74.58
	Multi-head + AT [19]	-	-	86.73	-	-	75.52
	Relation-Metric [20]	86.16	88.08	87.11	77.36	77.25	77.29
	SpERT [10]	88.99	89.59	89.28	77.77	79.96	78.84
	SPAN [9]	89.88	91.32	90.59	79.56	81.93	80.73
	TANL [21]	-	-	90.20	-	-	80.60
	REBEL [22]	-	-	-	-	-	81.70
	SSPC (ours)	<b>90.66</b>	<b>91.32</b>	<b>90.99</b>	<b>80.79</b>	<b>82.86</b>	<b>81.79</b>

#### 4.3. Comparison with the State-of-the-Art Model of Relation Extraction

In this part, we compare SSPC with several recent relation extraction models using prompts. Moreover, we validate the model's capabilities in two settings, including standard supervised training and few-shot learning.

- **ENT MARKER and TYP MARKER:** [39] uses prompt tuning for relation extraction. ENT MARKER injects special symbols to index the positions of entities. It is similar to prompting by introducing extra serial information to indicate the position of special tokens, i.e., named entities. TYP MARKER additionally introduces the type information of entities. It could be regarded as a type of template for prompts but requires additional annotation of type information.
- **PTR:** [3] proposes prompt tuning with rules (PTR) for many-class text classification and applies logic rules to construct prompts with several sub-prompts.
- **KnowPrompt:** [35] incorporates knowledge among relation labels into prompt-tuning for relation extraction and proposes a knowledge-aware prompt-tuning approach with synergistic optimization. It is the current state-of-the-art model on TACTED and TACREV in few-shot learning scenarios.

Table 4 shows our experimental results. The first column is the names of the baseline models. We use micro-averaged *F1* (excluding “no\_relation” type) for evaluation and treat a  $\langle S, R, O \rangle$  as correct when *S* and *O*'s boundaries are correct. From the results, we can see that our model performs well. In the standard supervised setting, *F1* increases by 0.3% on the TACRED dataset and 0.7% on the TACREV dataset. In the few-shot setting, *F1* increases by 0.3% on the TACRED dataset and 1.6% on the TACREV dataset on average. Note that when *K* is set to 32, the effect is the best, and *F1* increases by 3.0%. The experimental results prove that SSPC has fully adapted to the relation extraction task and is very effective in few-shot learning scenarios. Without introducing additional knowledge, SSPC still outperforms other robust baseline models. We owe these performance increases to our sub-prompts combination method.

**Table 4.** Our experimental results on the datasets TACRED and TACREV. F1 scores of various models with different sizes of training instances. The best results are bold.

Model	Standard Setting All Data	TACRED				Standard Setting All Data	TACREV			
		Few-Shot Setting					Few-Shot Setting			
		8	16	32	Mean	8	16	32	Mean	
ENT MARKER [39]	69.4	27.0	31.3	31.9	30.1	79.8	27.4	31.2	32.0	30.2
TYP MARKER [39]	71.0	28.9	32.0	32.4	31.1	80.8	27.6	31.2	32.0	30.3
PTR [3]	72.4	28.1	30.7	32.1	30.3	81.4	28.7	31.4	32.4	30.8
KnowPrompt [35]	72.4	32.0	<b>35.4</b>	36.5	34.6	82.4	32.1	33.1	34.7	33.3
SSPC (ours)	<b>72.7</b>	<b>32.7</b>	34.9	<b>37.0</b>	<b>34.9</b>	<b>83.1</b>	<b>32.7</b>	<b>34.2</b>	<b>37.7</b>	<b>34.9</b>

#### 4.4. Ablation Study

As shown in Table 5, we conduct ablation experiments to further evaluate the contribution of each part of the “ $\langle S, R, O \rangle$  Classification” in our proposed SSPC. “w/o (3-2)” denotes ablating the part of (3-2) in Figure 2, leaving only the part of (3-2) to filter the elements in the intermediate result set of the triple; “w/o (3-1)” denotes ablating the part of (3-1) in Figure 2, leaving only the part of (3-2) to filter the elements in the intermediate result set of the triple; “w/o (3-1) and (3-2)” denotes ablating the part of (3-1) and (3-2) in Figure 2, regarding all the elements in the intermediate result set as the actual triples in the sentence without filtering. We observe that both (3-1) and (3-2) are helpful for the joint extraction. When we ablate (3-1) and (3-2) simultaneously, *Recall* increases significantly, and *Precision* decreases significantly, indicating that we need these two parts to filter intermediate results.

**Table 5.** Ablations on the ADE dataset.

Method	Precision	Recall	F1
Full	80.79	82.86	81.79
w/o (3-2)	68.33	80.18	73.78
w/o (3-1)	73.84	81.25	77.37
w/o (3-1) and (3-2)	60.40	87.79	71.56

#### 4.5. $\langle S, R \rangle$ and $\langle R, O \rangle$ Extraction Inspection

Since our model can extract more fine-grained elements  $\langle S, R \rangle$  and  $\langle R, O \rangle$ , we also observed their extraction results on the ADE dataset. The experimental results are shown in Table 6. The *Precision* of  $\langle Effect, Adverse - Effect \rangle$  extraction reaches 85.27%, the *Recall* reaches 86.72%, and the *F1* reaches 85.99%; the *Precision* of  $\langle Adverse - Effect, Drug \rangle$  extraction reaches 96.05%, the *Recall* reaches 95.92%, and the *F1* reaches 95.99%. By analyzing the results, we can make the following observations. Firstly, SSPC can not only complete the entity recognition task, but also achieve more fine-grained extraction by integrating entity and relation information, which is critical for subsequent tasks. Secondly, SSPC has a solid ability to capture *Drug*-type entities. However, the extraction ability of *Effect*-type entities is unsatisfactory. We speculate that the pre-trained model has more and better knowledge of drugs.

**Table 6.** Detailed experimental results on the ADE dataset.

K	$\langle \text{Effect}, \text{Adverse} - \text{Effect} \rangle$			$\langle \text{Adverse} - \text{Effect}, \text{Drug} \rangle$		
	Precision	Recall	F1	Precision	Recall	F1
1	86.75	85.69	86.22	94.97	94.38	94.67
2	84.40	90.05	87.13	95.97	96.75	96.36
3	85.57	87.37	86.46	96.95	95.01	95.97
4	85.56	85.41	85.49	95.66	95.07	95.37
5	85.87	87.07	86.47	95.92	97.53	96.72
6	84.81	84.39	84.60	97.73	94.23	95.95
7	84.88	87.20	86.02	95.61	95.61	95.61
8	82.72	85.89	84.27	96.75	95.21	95.98
9	86.44	88.08	87.25	96.33	97.52	96.92
10	85.71	86.01	85.86	94.61	97.93	96.24
Mean	85.27	86.72	85.99	96.05	95.92	95.99

## 5. Conclusions

In this paper, we propose a new extraction paradigm. Based on it, we propose SSPC, an approach for Span-based Fine-Grained Entity-Relation Extraction via Sub-Prompts Combination. Experiments show that SSPC, under the guidance of the new paradigm, can comprehensively model the information fusion of entities and relations in the extraction process, extract all triples in sentences, and adapt to relation extraction. In future work, we will design more advanced model structures to extract information under the guidance of the new paradigm. We also plan to explore more advanced methods for stimulating knowledge in pre-trained language models.

**Author Contributions:** Conceptualization, N.Y.; Methodology, N.Y.; Formal analysis, N.Y.; Data curation, N.Y.; Writing—original draft, N.Y.; Writing—review & editing, N.Y., J.L. and Y.S.; Funding acquisition, J.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China (U21B2020, U1936216).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data used to support the findings of this study are available from the corresponding author upon request.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Zeng, D.; Liu, K.; Lai, S.; Zhou, G.; Zhao, J. Relation classification via convolutional deep neural network. In Proceedings of the COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, Dublin, Ireland, 23–29 August 2014; pp. 2335–2344.
- Zhou, P.; Shi, W.; Tian, J.; Qi, Z.; Li, B.; Hao, H.; Xu, B. Attention-based bidirectional long short-term memory networks for relation classification. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016; pp. 207–212.
- Han, X.; Zhao, W.; Ding, N.; Liu, Z.; Sun, M. PTR: Prompt Tuning with Rules for Text Classification. *AI Open* **2022**, *3*, 182–192. [[CrossRef](#)]
- Miwa, M.; Bansal, M. End-to-End Relation Extraction using LSTMs on Sequences and Tree Structures. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016; pp. 1105–1116.
- Adel, H.; Schütze, H. Global Normalization of Convolutional Neural Networks for Joint Entity and Relation Classification. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 7–11 September 2017; pp. 1723–1729.

6. Katiyar, A.; Cardie, C. Going out on a limb: Joint extraction of entity mentions and relations without dependency trees. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 917–928.
7. Yamada, I.; Asai, A.; Shindo, H.; Takeda, H.; Matsumoto, Y. LUKE: Deep Contextualized Entity Representations with Entity-aware Self-attention. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 16–20 November 2020; pp. 6442–6454.
8. Zeng, X.; Zeng, D.; He, S.; Liu, K.; Zhao, J. 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, Melbourne, Australia, 15–20 July 2018; pp. 506–514.
9. Ji, B.; Yu, J.; Li, S.; Ma, J.; Wu, Q.; Tan, Y.; Liu, H. Span-based joint entity and relation extraction with attention-based span-specific and contextual semantic representations. In Proceedings of the 28th International Conference on Computational Linguistics, Barcelona, Spain, 8–13 December 2020; pp. 88–99.
10. Eberts, M.; Ulges, A. Span-Based Joint Entity and Relation Extraction with Transformer Pre-Training. In Proceedings of the ECAI 2020, Santiago de Compostela, Spain, 29 August–8 September 2020; pp. 2006–2013.
11. Li, X.; Yin, F.; Sun, Z.; Li, X.; Yuan, A.; Chai, D.; Zhou, M.; Li, J. Entity-Relation Extraction as Multi-Turn Question Answering. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 1340–1350.
12. Zhao, T.; Yan, Z.; Cao, Y.; Li, Z. Asking effective and diverse questions: A machine reading comprehension based framework for joint entity-relation extraction. In Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, Yokohama, Japan, 7–15 January 2021; pp. 3948–3954.
13. Chen, Z.; Guo, C. A pattern-first pipeline approach for entity and relation extraction. *Neurocomputing* **2022**, *494*, 182–191. [[CrossRef](#)]
14. Takanobu, R.; Zhang, T.; Liu, J.; Huang, M. A hierarchical framework for relation extraction with reinforcement learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 7072–7079.
15. Xie, C.; Liang, J.; Liu, J.; Huang, C.; Huang, W.; Xiao, Y. Revisiting the Negative Data of Distantly Supervised 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, Online, 5–6 August 2021; pp. 3572–3581.
16. Li, F.; Zhang, Y.; Zhang, M.; Ji, D. Joint Models for Extracting Adverse Drug Events from Biomedical Text. In Proceedings of the IJCAI, New York, NY, USA, 9–15 July 2016; Volume 2016, pp. 2838–2844.
17. Li, F.; Zhang, M.; Fu, G.; Ji, D. A neural joint model for entity and relation extraction from biomedical text. *BMC Bioinform.* **2017**, *18*, 1–11. [[CrossRef](#)] [[PubMed](#)]
18. Bekoulis, G.; Deleu, J.; Demeester, T.; Develder, C. Joint entity recognition and relation extraction as a multi-head selection problem. *Expert Syst. Appl.* **2018**, *114*, 34–45. [[CrossRef](#)]
19. Bekoulis, I.; Deleu, J.; Demeester, T.; Develder, C. Adversarial training for multi-context joint entity and relation extraction. In Proceedings of the EMNLP2018, the Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 1–7.
20. Tran, T.; Kavuluru, R. Neural metric learning for fast end-to-end relation extraction. *arXiv* **2019**, arXiv:1905.07458.
21. Paolini, G.; Athiwaratkun, B.; Krone, J.; Ma, J.; Achille, A.; Anubhai, R.; dos Santos, C.N.; Xiang, B.; Soatto, S. Structured Prediction as Translation between Augmented Natural Languages. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 26–30 April 2020.
22. Cabot, P.L.H.; Navigli, R. REBEL: Relation extraction by end-to-end language generation. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2021, Punta Cana, Dominican Republic, 16–20 November 2021; pp. 2370–2381.
23. Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. Improving language understanding by generative pre-training. *OpenAI* **2018**. Available online: <https://www.cs.ubc.ca/~amuham01/LING530/papers/radford2018improving.pdf> (accessed on 1 December 2022)
24. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the NAACL-HLT, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186.
25. Liu, X.; Zheng, Y.; Du, Z.; Ding, M.; Qian, Y.; Yang, Z.; Tang, J. GPT understands, too. *arXiv* **2021**, arXiv:2103.10385.
26. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **2020**, *21*, 1–67.
27. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
28. Schick, T.; Schütze, H. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv* **2020**, arXiv:2001.07676.
29. Rubin, O.; Herzig, J.; Berant, J. Learning to retrieve prompts for in-context learning. *arXiv* **2021**, arXiv:2112.08633.
30. Li, J.; Tang, T.; Nie, J.Y.; Wen, J.R.; Zhao, W.X. Learning to Transfer Prompts for Text Generation. *arXiv* **2022**, arXiv:2205.01543.

31. Kasahara, T.; Kawahara, D.; Tung, N.; Li, S.; Shinzato, K.; Sato, T. Building a Personalized Dialogue System with Prompt-Tuning. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Student Research Workshop, Online, 10–15 July 2022; pp. 96–105.
32. Cui, L.; Wu, Y.; Liu, J.; Yang, S.; Zhang, Y. Template-based named entity recognition using BART. *arXiv* **2021**, arXiv:2106.01760.
33. Lee, D.H.; Agarwal, M.; Kadakia, A.; Pujara, J.; Ren, X. Good examples make A faster learner: Simple demonstration-based learning for low-resource NER. *arXiv* **2021**, arXiv:2110.08454.
34. Ma, R.; Zhou, X.; Gui, T.; Tan, Y.; Zhang, Q.; Huang, X. Template-free prompt tuning for few-shot NER. *arXiv* **2021**, arXiv:2109.13532.
35. Chen, X.; Zhang, N.; Xie, X.; Deng, S.; Yao, Y.; Tan, C.; Huang, F.; Si, L.; Chen, H. Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction. In Proceedings of the ACM Web Conference 2022, Lyon, France, 25–29 April 2022; pp. 2778–2788.
36. Gurulingappa, H.; Rajput, A.M.; Roberts, A.; Fluck, J.; Hofmann-Apitius, M.; Toldo, L. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. *J. Biomed. Inform.* **2012**, *45*, 885–892. [[CrossRef](#)] [[PubMed](#)]
37. Zhang, Y.; Zhong, V.; Chen, D.; Angeli, G.; Manning, C.D. Position-aware attention and supervised data improve slot filling. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 7–11 September 2017.
38. Alt, C.; Gabryszak, A.; Hennig, L. TACRED Revisited: A Thorough Evaluation of the TACRED Relation Extraction Task. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, online, 5–10 July 2020; pp. 1558–1569.
39. Zhou, W.; Chen, M. An Improved Baseline for Sentence-level Relation Extraction. In Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing, Minneapolis, MN, USA, 20–21 November 2022; pp. 161–168.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.