electronics

MDPI

Article

RECA: Relation Extraction Based on Cross-Attention Neural Network

Xiaofeng Huang, Zhiqiang Guo, Jialiang Zhang *, Hui Cao and Jie Yang

School of Information Engineering, Wuhan University of Technology, Wuhan 430070, China; 259079@whut.edu.cn (X.H.); guozhiqiang@whut.edu.cn (Z.G.); iehuicao@whut.edu.cn (H.C.); jieyang@whut.edu.cn (J.Y.)
* Correspondence: zjliang@whut.edu.cn

Abstract: Extracting entities and relations, as a crucial part of many tasks in natural language processing, transforms the unstructured text information into structured information and provides corresponding data support for knowledge graph (KG) and knowledge vault (KV) construction. Nevertheless, the mainstream relation-extraction methods, the pipeline method and the joint method, ignore the dependency between the subject entity and the object entity. This work introduces a pre-trained BERT model and a dilated gated convolutional neural network (DGCNN) as an encoder to distinguish the long-range semantics representation from the input sequence. In addition, we propose a cross-attention neural network as a decoder to learn the importance of each subject word for each word of the input sequence. Experiments were undertaken with two extensive datasets, the New York Times Corpus (NYT) and WebNLG Corpus, and showed that our model performs significantly better than the CasRel model, outperforming the baseline by 1.9% and 0.7% absolute gain in terms of F1-score.

Keywords: cross-attention neural network; dilated gated convolutional neural network; joint method; relation extraction

Citation: Huang, X.; Guo, Z.; Zhang, J.; Cao, H.; Yang, J. RECA: Relation Extraction Based on Cross-Attention Neural Network. *Electronics* **2022**, *11*, 2161. https://doi.org/10.3390/electronics11142161

Academic Editors: Matúš Pleva, Piotr Szczuko, Daniel Hládek and Andrej Zgank

Received: 27 May 2022
Accepted: 8 July 2022
Published: 11 July 2022

1. Introduction

Relation extraction aims at converting unstructured text information into structured information and it is a fundamental task for large-scale knowledge graph and knowledge vault construction [1]. It provides essential data services for natural language processing, including information extraction, question answering, and semantic analysis.

The objective of relation extraction is the extraction of relation triplets consisting of a subject, object, and the relations between them. Relation triplets are shown as (subject, relation, object) or (s, r, o). Early work on relation extraction generally applied a pipeline method [2,3] dividing relation extraction into two subtasks named entity recognition (NER) and relation classification (RC). This approach consists of recognizing all entities from sentences and extracting relations from each entity pair, which means that the error propagation problem may be encountered. A joint method for relation extraction [4,5] was proposed to solve this problem. This method directly detects the complete relational triplets, and it includes feature-based models. Nevertheless, most relation extraction approaches based on joint models require complex semantic representation, which means traditional manual feature construction is not applicable. Recently, neural network-based models [6,7] have been applied to extract sentence representation and complete relation extraction, achieving outstanding performance.

Though the joint method for relation extraction has achieved considerable success, most existing studies have ignored the overlapping problem, where a sentence contains multiple relational triplets. Figure 1 shows the overlapping problem in relation extraction as proposed by Zeng [8]. An EntityPairOverlap (EPO) problem means multiple relations between an entity pair. A SingleEntityOverlap (SEO) problem means an entity has relations

with different entities in a sentence. It is hard to identify all triplets from a sentence with the traditional joint method when the overlapping problem occurs. To tackle these issues, related studies have proposed a sequence-to-sequence (Seq2Seq) model to generate all the relational triplets in a sentence [8,9]. Later, a graph convolutional neural network [10] was applied to construct graphs of entity pairs and detect all relations between each entity in a sentence.

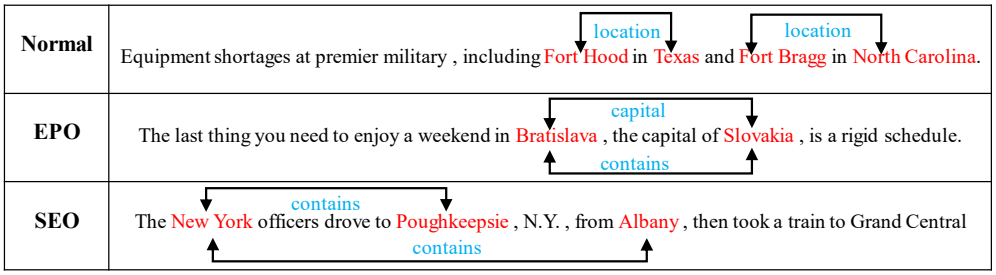

Figure 1. Examples of normal, EPO, and SEO overlapping problems.

Although great success has been achieved in previous work on extracting entities and relations, most relation extraction models ignore the fact that subjects contain rich semantic information on objects and relations that has immediate relevance to the relation extraction. Moreover, Wei et al. [11] and Sun et al. [12] have proven that fusing the relevant representations into context representation enables the enhancement of model performance in relation extraction, and the attention model has proven its effectiveness in representation fusion [13]. Furthermore, Lai et al. [14] have utilized an attention model to extract relational triplets, further proving its effectiveness in relation extraction.

In this paper, we propose a joint method for relation extraction based on a cross-attention neural network (RECA) to exploit the relevance of each subject word for each word of the input sequence. A pre-trained BERT model and a dilated gated convolutional neural network (DGCNN) are employed to encode the input sequence into a semantic representation. Next, the relation extraction task is divided into two subtasks: (1) the subject taggers are modeled through encoder representation and (2) a cross-attention mechanism is used to fuse the subject and sentence representations. Then, the relation-object taggers are modeled through the fusion representation to recognize possible relations and objects. Our model considers the sentence representation and uses the subject's semantic information to identify objects and relations.

The contributions of this paper can be summarized as follows:

1. We introduce a dilated gated convolutional neural network model, which has the advantage of being able to learn long-range information, instead of traditional convolution. In addition, we propose residual gated linear units as an activation function to improve model performance.
2. We propose a cross-attention neural network for the decoder to learn the dependence between subject information and relation-object taggers. This combines subject representation with sentence representation to detect relevant objects and relations.
3. We describe a series of experiments in which we compared the proposed model with the CasRel [11] model and show that our model achieved 1.9% and 0.7% absolute gain in terms of F1-score for two datasets. The ablation experiment demonstrates that each part we propose improves network performance.

2. Related Works

Extraction of relational triplets from unstructured sentences has always been a fundamental task for information extraction. Early work divided relation extraction into entity recognition and relation classification, which were realized by the pipeline method [3,4]. This method is prone to propagation errors and neglects the relevance connecting the two tasks. To address these problems, the joint method was proposed. Zheng et al. [15] pro-

posed a unified tagging scheme that tags each word's position and relation information in a sentence. This tagging method converts the relation extraction problem into an end-to-end tagging problem and enables the model to jointly extract complete relational triplets.

Early work on joint methods was based on complex feature extraction. Recently, language models such as Embeddings from Language Models (ELMo) [16] and Bidirectional Encoder Representation from Transformers (BERT) [17] have been proposed, which have further improved the capabilities of semantic representation extraction. Several models based on pre-trained language models [18,19] have also been proposed to extract relational triplets and significantly improve model performance.

Though joint methods have achieved great success, most neglect the overlapping problem. Zeng et al. [8] proposed two patterns for overlapping problems and utilized a sequence-to-sequence (Seq2Seq) model with a copy mechanism to generate all relational triplets. Ye et al. [9] proposed a generative transformer to identify relational triplets and utilized triplet contrastive learning to enhance model performance. In addition, Fu et al. [10] addressed the overlapping problem by treating words in sentences as nodes of a graph and relations between the entities as edges of a graph and then introducing a graph convolutional neural network to extract relational triplets. Wei et al. [11] proposed the CasRel framework, an end-to-end cascade binary tagging framework. They designed a subject tagger to detect all subjects directly and then modeled relations as a function that maps subjects to objects. The CasRel framework extracts the subject's first and end token representations to influence object and relation identification. Sun et al. [12] proposed the PMEI framework, which enhances model performance by providing a novel framework to control information flow. The PMEI framework utilizes the representation of relation extraction to enhance the performance of entity recognition and applies the entity recognition representation to improve relation extraction. CasRel and PMEI demonstrate the effectiveness of representation fusion in relation extraction. Lai et al. [13] proposed the RMAN framework, introducing a multi-head attention model to control the representation flow and obtain the final sentence representation. The RMAN framework demonstrates the effectiveness of the attention model in relation extraction.

In this work, we propose RECA for relation extraction. Our model consists of a pre-trained BERT model and DGCNN encoder module, a subject tagging module, a cross-attention neural network module, and a relation-object tagging module. We describe the details of our model below.

3. Model

In this section, we describe each module of the RECA model in detail. Figure 2 illustrates the overview of our model. The goal of relation extraction is to extract all triplets $\{s, r, o\}$ from sentences. Considering the overlapping problem, we divide relation extraction into two subtasks: a subject tagger task and a relation-object tagger task. We encode an input sentence as a semantic representation vector $H_E = \{h^1, \ldots, h^L\}$. Then, we use a subject tagger to identify the start positions *s_start* and the end positions *s_end* of all subjects given H_E. In addition, we extract subject representation $S = \left\{h^{s\_start}, \ldots, h^{s\_end}\right\}$ based on *s_start* and *s_end*. We use a cross-attention neural network to achieve fusion representation H_{CR} based on subject representation and sentence representation. We deploy a relation-object tagger to identify the start positions *o_start* and end positions *o_end* of all objects for a predefined relation r based on H_{CR}.

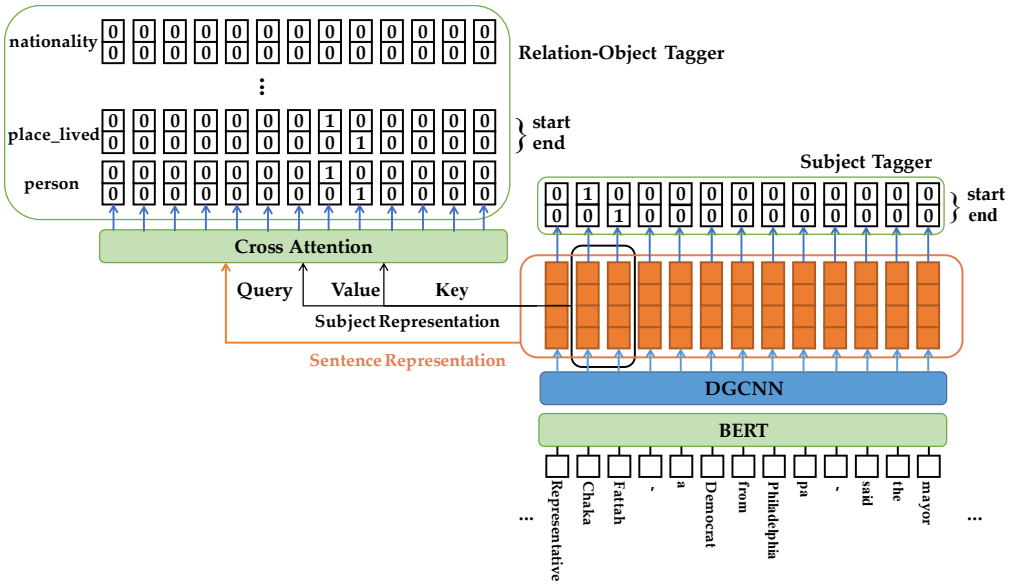

Figure 2. An overview of the RECA model. In this example, "Chaka Fattah" is the subject of this sentence. We tag "1" for "Chaka" in the start position vector and tag "0" for the other words. Similarly, we tag "1" for the word "Fattah" in the end position vector. The RECA model uses an encoder module to convert words into sentence representations and detects the position information of subjects. There are two relational triplets for the subject "Chaka Fattah" in this sentence. Therefore, we tag "1" for the words "Philadelphia" and "pa" in the "place_lived" and "person" relation-object start- and end-position vector. Note that we tag "0" for other relation-object position vectors. The RECA model decodes sentence representation and subject representation and detects the position information of the object for each relation in the relation-object tagger.

3.1. Encoder

3.1.1. BERT

We utilize a pre-trained BERT model and a dilated gated convolution neural network as the encoder module. The BERT model consists of a multi-layer bidirectional transformer [20]. It relies on a self-attention mechanism to determine global dependencies instead of a recurrent model. The BERT model has exhibited excellent performance in many natural language process tasks [21].

The BERT model makes it possible to convert sentences into a vector by embedding subwords into a matrix, and it then feeds the embedding vector into multi-layer transformer blocks and returns context representation $H_{BERT} = \{h^1_{BERT}, \ldots, h^L_{BERT}\}$.

3.1.2. Dilated Gated Convolutional Neural Network

As shown in Figure 3, considering that convolutional networks can represent large context sizes and extract hierarchical features over larger contexts with more abstract features, we use a dilated gated convolutional neural network model to exploit contextual representation efficiently. Inspired by Jonas et al. [22], we utilize dilated convolution [23] for further learning of long-range information instead of traditional convolution. This is undertaken by inserting "holes" that do not participate in the convolution operation between each pixel in a kernel, which supports an exponentially expanding receptive field. In addition, there is a degradation problem in deep learning models with increasing trainable parameters. Therefore, we use a residual mechanism with gated linear units [24] that can simultaneously address the above two problems. We utilize the sigmoid as the activation function since it can map the input representation to $(0, 1) \in \mathbb{R}$, enabling control of the information flow [24]. In contrast to previous work, we propose a residual mechanism unit focused on the generation of the gated unit through a dilated convolutional network. This controls the degree of retention for the output representation and the degree

of forgetting for the input representation. Additionally, we add two parts of representations to address the degradation problem.

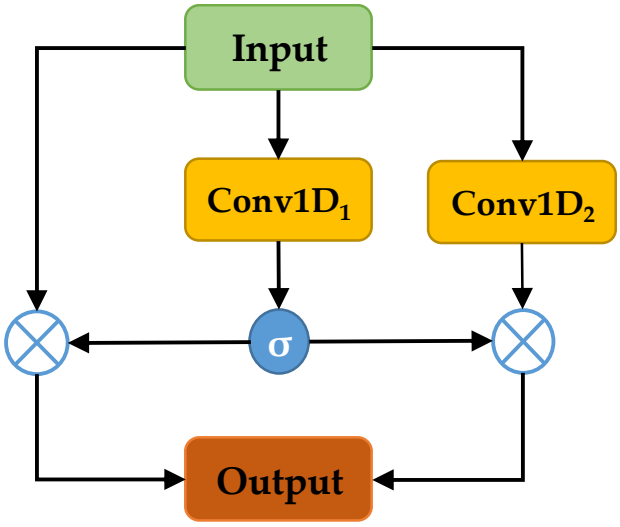

Figure 3. An overview of the residual gated linear unit.

In this work, each dilated convolution kernel is parameterized as $W \in \mathbb{R}^{2d \times kd}$ and $b_w \in \mathbb{R}^{2d}$, where d is the size of the input sequence and k is the kernel size of the dilated convolution. The output of the dilated convolution is $h_{DCNN} \in \mathbb{R}^{2d}$. We use a residual gated linear unit as an activation function, and the detailed operations can be formulated as:

$$h_{DCNN} = [Conv1D_1, Conv1D_2] \tag{1}$$

$$\sigma = Sigmoid(Conv1D_2) \tag{2}$$

$$h_{DGCNN} = h_{BERT} \times (1 - \sigma) + Conv1D_1 \times \sigma \tag{3}$$

We divide the dilated convolutional network outputs h_{DCNN} into two parts: $Conv1D_1$, $Conv1D_2 \in \mathbb{R}^{d_{model}}$. We use the gate unit σ to control the path through which information flows in the network [24]. In our work, the gate unit σ can decide which DCNN representation should be propagated through the hierarchy of layers and which BERT representation should be forgotten by multiplying with these representations. We add two parts of relevant representations as the DGCNN output $h_{DGCNN} \in \mathbb{R}^d$.

3.2. Decoder

3.2.1. Subject Tagger

In this work, we refer to the tagging scheme for the CasRel framework [11]. The tag "1" represents a subject's start and end position, and the tag "0" represents other positions. We use two binary classifiers to identify the probability of each word as the start and end positions of a subject given the contextual representation H_E, which consists of the DGCNN output h_{DGCNN}. The detailed operations are as follows:

$$p^i_{s\_start} = Sigmoid(W_{start} h^i_E + b_{start}) \tag{4}$$

$$p^i_{s\_end} = Sigmoid(W_{end} h^i_E + b_{end}) \tag{5}$$

where $p^i_{s\_start}$ and $p^i_{s\_end}$ represent the probability of the ith token in the sequence as the start and end positions of the subject. The subject tagger optimizes the following likelihood function to identify the span of the subject s given encoder representation H_E:

$$p_\theta(s|H_E) = \prod_{t\in\{s\_start, s\_end\}} \prod_{i=1}^{L} (p_i^t)^{I\{y_i^t=1\}} (1-p_i^t)^{I\{y_i^t=0\}} \tag{6}$$

where L represents the length of the sequence and $y_i^{s\_start}$ and $y_i^{s\_end}$ are the tags for the start and end positions for ith token in the sequence. $I\{z\} = 1$ if z is true and 0 otherwise.

3.2.2. Cross-Attention Neural Network

Considering the relevance connecting the subject and object, we tend to use the subject tagger representation to enhance relation-object tagger performance with the attention model. The attention model aims to learn attention weights through an additional deep neural network and normalizes them using the SoftMax activation function. This model is applied to influence encoder representation and thus improve model performance. Previous work on the attention model utilized the dot product function [20], biased general function [25], and other functions to generate attention weights. Such work applied addition, multiplication, and concatenation to influence target representation and only consider the correlation within a single sequence rather than between multiple input sequences. Related work on attention mechanisms focus on self-attention, considering only the internal correlation within the input sequence, which does not apply in our model. Therefore, we use a cross-attention neural network that calculates the correlation between the two input sequences and generates the attention representation that represents the importance of each sequence token for each token of another sequence.

Moreover, we utilize a cross-attention mechanism to fuse the subject representation s and sentence representation H_E and enhance the relation-object tagger performance. This mechanism takes two parts of representations into account and generates a new attention representation. We generate the subject representation $S = \left\{h_E^{s\_start}, \ldots, h_E^{s\_end}\right\}$ based on the encoder output H_E and subject position information s_start and s_end. The cross-attention neural network performs the following operations on two parts of representations:

$$K = relu(W_{key}S + b_{key}) \tag{7}$$

$$V = relu(W_{value}S + b_{value}) \tag{8}$$

$$Q = relu(W_{query}S + b_{query}) \tag{9}$$

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_{CR}}})V \tag{10}$$

where W is a trainable matrix that does not share parameters and d_{CR} represents the dimension of the matrix V, K, Q. Inspired by the self-attention mechanism [20], we generate a cross-attention matrix to represent the importance of each subject token for each token of the input sentence. The cross-attention neural network generates the query matrix Q from the sentence representation H_E. The key matrix K and the value matrix V are generated from the subject representation S. We compute the cross-attention representation matrix $Attention = \left\{A^1, \ldots, A^L\right\}$, where $A^i \in \mathbb{R}^{d_{CR}}$, based on the scaled dot-product attention mechanism. Our model employs a residual mechanism [26] and layer normalization [27] in a cross-attention neural network to alleviate the over-fitting phenomenon. The detailed operations are as follows:

$$h_f^i = gelu(W_f A^i + b_f) \tag{11}$$

$$h_{CR}^i = LayerNorm(h_E^i + h_f^i) \tag{12}$$

where $h_f^i \in \mathbb{R}^{d_{model}}$ is the feedforward neural network output and $h_{CR}^i \in \mathbb{R}^{d_{model}}$ is the cross-attention neural network output. Notably, we utilize the approximate Gaussian Error Linear Unit (GELU) function instead of the Rectified Linear Unit (ReLU) as an activation function to improve the model generalization.

3.2.3. Relation-Object Tagger

In our model, the relation-object tagger detects objects and relations according to the cross-attention neural network output H_{CR}^i. The relation-object tagger consists of multiple object taggers, which are the same as subject taggers. Each object tagger identifies corresponding objects for a predefined relation. The detailed operations are as follows:

$$p_{o\_start}^i = Sigmoid(W_{start}^r h_{CR}^i + b_{start}^r) \tag{13}$$

$$p_{o\_end}^i = Sigmoid(W_{end}^r h_{CR}^i + b_{end}^r) \tag{14}$$

where r represents the detection of the rth predefined relation and $p_{o\_start}^i$ and $p_{o\_end}^i$ represent the probability of identifying the ith token in the sequence as the start and end positions of objects. Similarly to the subject tagger, the following likelihood function is optimized by the relation-object tagger for relation r to detect the span of an object o given the cross-attention representation H_{CR}:

$$p_\theta(s|H_{CR}) = \prod_{t\in\{o\_start, o\_end\}} \prod_{i=1}^{L} (p_i^t)^{I\{y_i^t=1\}} (1-p_i^t)^{I\{y_i^t=0\}} \tag{15}$$

where $y_i^{o\_start}$ and $y_i^{o\_end}$ are the tags of the start and end positions for ith token in the sequence. We use the log-likelihood objective function $\mathcal{L}$ based on Equations (6) and (15). The detailed operation is as follows:

$$\mathcal{L} = \sum_{j=1}^{|D|} \left[\begin{array}{l} \sum\limits_{s\in T_j} \log p_\theta(s|H_E) + \sum\limits_{r\in T_j|s} \log p_\theta(o|H_{CR}) + \\ \sum\limits_{r\notin T_j|s} \log p_\theta(o_\varnothing|H_{CR}) \end{array} \right] \tag{16}$$

where D represents datasets, T_j represents the jth sentence in datasets, and $r \in T_j|s$ represents that the subject s in sentence T_j contains relation r. We optimize our model by maximizing function $\mathcal{L}$ through an Adam stochastic gradient descent [28].

4. Experiment

4.1. Experiment Datasets and Evalution Metrics

This section describes the evaluation of the RECA model using two public datasets: NYT [29] and WebNLG [30]. A distance supervision method was used to generate the original NYT dataset, which contains more than one million sentences covering 24 predefined relations. The original WebNLG dataset covered 246 predefined relations and was adapted for relation extraction by Zeng et al. [8]. The sentences in both datasets can be used to support the evaluation of our model's performance in tackling the overlapping problem and multiple-triplet problem. For a series of additional experiments, we split the sentences in the two datasets into three categories based on the types of overlapping problems. We further split the sentences into five categories based on the numbers of triplets in the sentences. Additionally, we split the sentences into two categories, long and short, based on the lengths of the sentences, where the sentences with more than 50 words were considered long sentences and the others short sentences. The statistics for the two datasets are shown in Table 1.

Table 1. Statistics for datasets. Note that a sentence can belong to both the EPO class and the SEO class.

Category		NYT		WebNLG	
		Training	**Testing**	**Training**	**Testing**
Overlap	Normal	37,013	3266	1596	246
	EPO	9782	978	227	26
	SEO	14,735	1297	3406	457
Number	n = 1	36,868	3244	1716	266
	n = 2	12,058	1045	1264	171
	n = 2	3663	312	1043	131
	n = 4	2618	291	648	90
	n $\geq$ 5	988	108	348	45
Length	Short	45,821	4054	4882	667
	Long	10,374	946	137	36
	ALL	56,195	5000	5019	703

Following previous work, we stipulated that the extracted relation triplets would only be considered correct when both entities (subject and object) and their relation were correct. For a fair comparison, we used standard micro-precision (Prec.), micro-recall (Rec.), and micro-F1-score (F1) as the metrics to evaluate model performance.

4.2. Setting Training Parameters

We used a pre-trained BERT model (BERT-Base, Cased) with default hyperparameters for fine-tuning, and the dimension of the hidden state d_{BERT} was 768. The dimension of the dilated convolution output d_{model} was 768. Note that the dimension in the cross-attention neural network d_{CR} was 512. We set the dimension of the feedforward neural network as 768 and used an Adam stochastic gradient descent to optimize our model. The learning rate in the pre-trained BERT model was 1×10^{-5} and the learning rate in the DGCNN and cross-attention neural network was 5×10^{-5}. We optimized our model with the batch size as 5 and introduced an early stopping mechanism when the F1-score in the validation set did not improve for 10 consecutive epochs. The threshold of both the start and end positions was 0.5.

4.3. Experimental Result

We compared the RECA model with state-of-the-art models from recent years to assess its performance, including NovelTagging [15], CopyR [8], GraphRel [10], CopyR$_{RL}$ [31], CasRel [11], PMEI [12], RMAN [14], and CGT [9]. Note that we used a pre-trained BERT model as part of the encoder in the RECA model for better performance. To further verify the cross-attention neural network performance and evaluate the pre-trained BERT model's impact, we used RECA$_{BiLSTM}$, which utilizes bi-directional long short-term memory (BiLSTM), instead of the pre-trained BERT model as part of the encoder. For a fair comparison, we re-implemented CasRel$_{BiLSTM}$ by replacing the pre-trained BERT model with BiLSTM. Note that RECA$_{BiLSTM}$ and CasRel$_{BiLSTM}$ utilize a trainable embedding layer with random initialization. The performance comparison for the RECA, RECA$_{BiLSTM}$, and previous state-of-the-art models is shown in Table 2. Additionally, we conducted a series of ablation experiments to evaluate the effectiveness of each module we proposed. We used RECA$_{DGCNN}$, which only utilizes DGCNN, as part of the encoder and RECA$_{CA}$, which only utilizes a cross-attention neural network, as part of the decoder. The performances of the RECA$_{DGCNN}$ and RECA$_{CA}$ models with the two datasets are shown in Table 3.

Table 2. Experimental results for different models and the NYT and WebNLG datasets. Our re-implementation is marked by *. The best scores are in bold font.

Method	NYT			WebNLG		
	Prec.	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
NovelTagging [15]	62.4	31.7	42.0	52.5	19.3	28.3
CopyR$_{OneDecoder}$ [8]	59.4	53.1	56.0	32.2	28.9	30.5
CopyR$_{MultiDecoder}$ [8]	61.0	56.6	58.7	37.7	36.4	37.1
GraphRel$_{1p}$ [10]	62.9	57.3	60.0	42.3	39.2	42.9
GraphRel$_{2p}$ [10]	63.9	60.0	61.9	44.7	41.1	42.9
CopyR$_{RL}$ [31]	77.9	67.2	72.1	63.3	59.9	61.6
CasRel * $_{BiLSTM}$	79.4	68.8	73.6	89.6	78.4	83.6
CasRel [11]	89.7	89.5	89.6	**93.4**	90.1	91.8
PMEI [12]	90.5	89.8	90.1	91.0	92.9	92
RMAN [14]	87.1	83.8	85.4	83.6	85.3	84.5
CGT [9]	**94.7**	84.2	89.1	92.9	75.6	83.4
RECA$_{BiLSTM}$	78.9	76.5	77.6	91.3	84.5	87.8
RECA	91.2	**91.9**	**91.5**	90.9	**94.1**	**92.5**

Table 3. Results for ablations experiments with the NYT and WebNLG datasets. The best scores are in bold font.

Method	NYT			WebNLG		
	Prec.	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
CasRel [11]	89.7	89.5	89.6	**93.4**	90.1	91.8
RECA	**91.2**	91.9	**91.5**	90.9	**94.1**	**92.5**
RECA$_{DGCNN}$	89.8	90.7	90.3	91.7	92.1	91.9
RECA$_{CA}$	90.3	**92.4**	91.3	92.5	92.0	92.3

We split the sentences in the NYT and WebNLG datasets into three categories based on different overlapping problems. We experimented on extracting triplets from these three types of sentences to evaluate the model performance in tackling the overlapping problem. The performance comparison of the RECA model and the previous models for the overlapping problem is shown in Figure 4a,b. Note that most sentences contained multiple relational triplets. To verify the performance of the RECA model in tackling multiple triplets, we classified the sentences according to the number of triplets in a sentence and conducted an extended experiment with the sentences containing multiple triplets. The comparison between our model and the previous models in tackling the overlapping problem is shown in Table 4. To further evaluate the contribution of the DGCNN in particular, we classified the sentences based on the lengths of sentences and assessed the performances of the CasRel, RECA$_{DGCNN}$, and RECA models in tackling long sentences. The detailed results are shown in Figure 5.

Table 4. F1-scores for the extraction of relational triplets from sentences with different numbers of triplets. The best scores are in bold font.

Method	NYT					WebNLG				
	n = 1	**n = 2**	**n = 3**	**n = 4**	**n $\geq$ 5**	**n = 1**	**n = 2**	**n = 3**	**n = 4**	**n $\geq$ 5**
CopyR$_{OneDecoder}$ [8]	66.6	52.6	49.7	48.7	20.3	65.2	33.0	22.2	14.2	13.2
CopyR$_{MultiDecoder}$ [8]	67.1	58.6	52.0	53.6	30.0	59.2	42.5	31.7	24.2	30.0
GraphRel$_{1p}$ [10]	69.1	59.5	54.4	53.9	37.5	63.8	46.3	34.7	30.8	29.4
GraphRel$_{2p}$ [10]	71.0	61.5	57.4	55.1	41.1	66.0	48.3	37.0	32.1	32.1
CasRel [11]	88.2	90.3	91.9	94.2	83.7	**89.3**	90.8	94.2	92.4	90.9
RECA	**89.5**	**92.1**	**93.3**	**95.8**	**90.4**	89.1	**92.1**	**94.8**	**93.3**	**91.3**

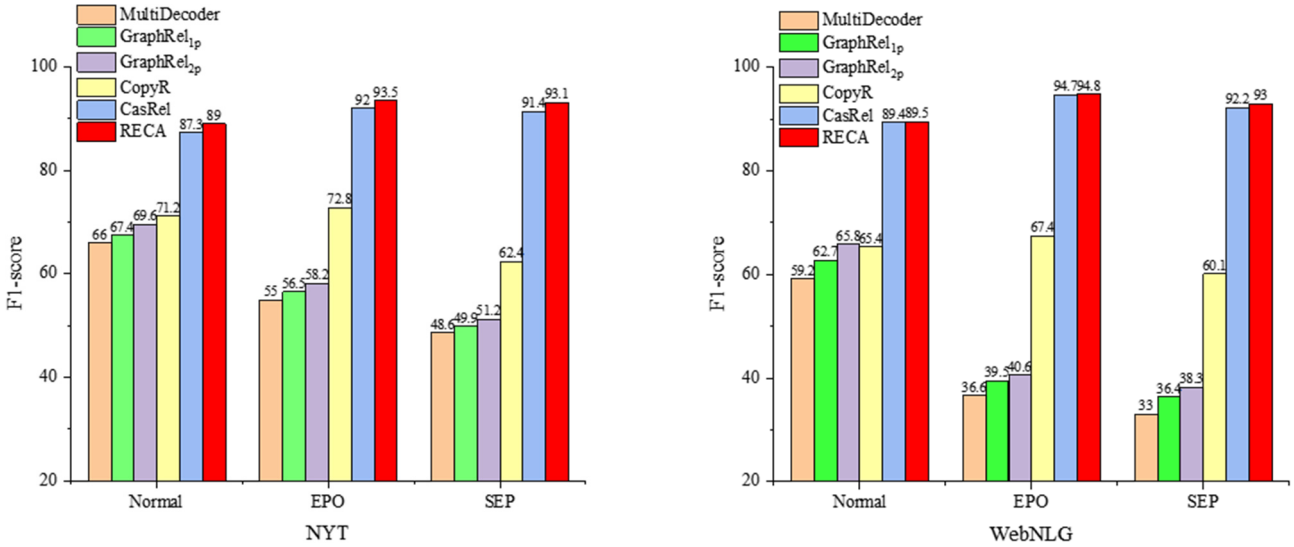

Figure 4. (**a**) The F1-score for the extraction of relational triplets from sentences with different overlapping problems from the NYT dataset and (**b**) the F1-score for the extraction of relational triplets from sentences with different overlapping problems from the WebNLG dataset.

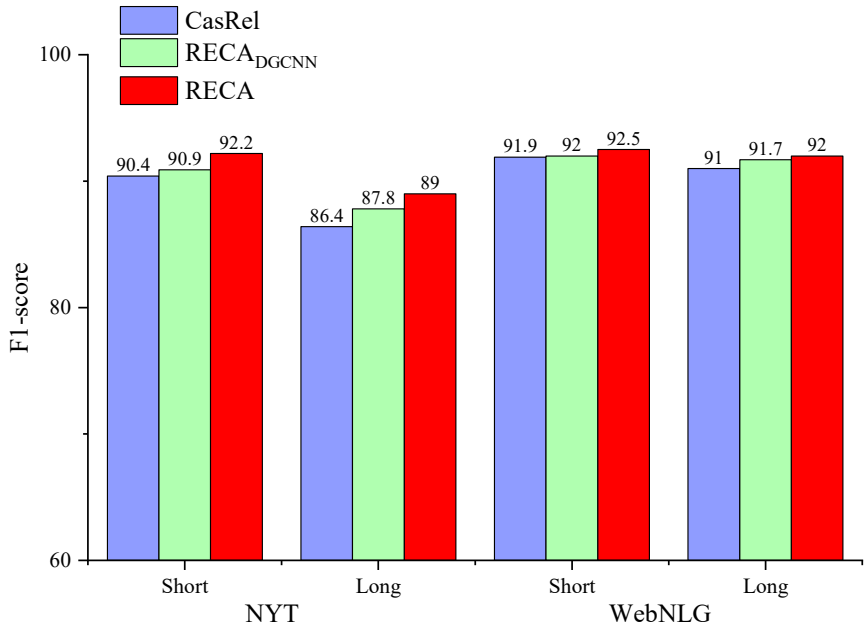

Figure 5. Result for long-sentence experiments with the NYT and WebNLG datasets.

5. Discussion

The RECA model outperformed all baseline models in terms of F1-score and achieved encouraging 1.9% and 0.7% improvements over CasRel for the NYT and WebNLG datasets, respectively. Moreover, the performance of $RECA_{BiLSTM}$ was also more competitive than most previous models except for CasRel. Compared with $CasRel_{BiLSTM}$, $RECA_{BiLSTM}$ achieved encouraging 4% and 4.2% improvements in the F1-score for the NYT and WebNLG datasets, respectively. The above experiments show that the RECA model performs far better than previous models. In the ablation experiment, we observed that $RECA_{DGCNN}$ and $RECA_{CA}$ outperformed CasRel. The $RECA_{CA}$ model, which only used a cross-attention neural network, achieved encouraging 1.7% and 0.5% improvements in the F1-score for NYT and WebNLG datasets compared to CasRel, proving the effectiveness of the cross-attention neural network in relation extraction. Notably, as shown in Table 2, the F1-score

for our model showed the most significant improvement when we utilized two networks simultaneously.

Moreover, we conducted supplementary experiments on the extraction of relational triplets from different types of sentences to further evaluate the capability of the RECA model in tackling the overlapping problem. As shown in Figure 4a,b, the RECA model outperformed all previous models in tackling sentences with overlapping problems. Compared to CasRel, the F1-scores for the RECA model increased by 1.9% and 0.8% when extracting triplets from sentences with the SEO problem from the NYT and WebNLG datasets. The RECA model achieved encouraging 1.5% and 0.1% improvements in the F1-score compared to CasRel when extracting triplets from sentences with the EPO problem from the two datasets. The above results prove that the performance of the RECA model in tackling the overlapping problem was improved compared to previous models. Additionally, to validate the contribution of the DGCNN, we conducted supplementary experiments on tackling long sentences. As shown in Figure 5, the RECA_{DGCNN} and RECA models achieved 1.4% and 2.6% absolute F1-score improvements over CasRel for the NYT dataset, which proves the effectiveness of the DGCNN in tacking long sentences. However, it was difficult for us to evaluate the contribution of the DGCNN in tackling long sentences from the WebNLG dataset since it contains few long sentences. As shown in Table 4, compared with previous models, the RECA model was able to tackle sentences with multiple relational triplets well, especially sentences with more than five triplets, achieving encouraging 6.7% and 0.4% improvements in the F1-score.

6. Conclusions

This paper proposed a relation extraction model based on a cross-attention neural network (RECA) and evaluated the model performance with NYT and WebNLG datasets. We divided the relation extraction into subject tagger and relation-object tagger tasks. We used a pre-trained BERT model and a DGCNN as the encoder and a cross-attention neural network as the decoder. In the cross-attention neural network, we fused the subject representation and sentence representation and computed the attention representation, enhancing the relation-object tagger's performance. In addition, we validated the capabilities of our model in identifying all triplets from sentences with the overlapping problem, the multiple-triplets problem, and the long-sentence problem. The experimental results showed that our model outperformed previous models. We surmise that the attention representation generated by the cross-attention neural network enables significant enhancement of model performance in relation extraction and other natural language processing tasks. We will apply the cross-attention neural network to other natural language processing tasks involving representation fusion to validate its capabilities.

Author Contributions: X.H.: Conceptualization, methodology, software, validation, resources, formal analysis, visualization, and writing—original draft preparation. Z.G.: investigation and data curation. J.Z.: writing—review and editing. H.C.: supervision and project administration. J.Y.: funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Natural Science Foundation of China (grant number 51479159).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: MDPI Research Data Policies at https://github.com/beaverbee/CrossAttention (accessed on 7 July 2022).

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

References

1. Young, G.O. *Synthetic Structure of Industrial Plastics*, 2nd ed.; Peters, J., Ed.; McGraw-Hill: New York, NY, USA, 1964; Volume 3, pp. 15–64.
2. Chan, Y.S.; Roth, D. Exploiting syntactico-semantic structures for relation extraction. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, OR, USA, 19–24 June 2011; Volume 1, pp. 551–560.
3. Zelenko, D.; Aone, C.; Richardella, A. Kernel methods for relation extraction. *J. Mach. Learn. Res.* **2002**, *10*, 71–78.
4. Li, Q.; Ji, H. Incremental joint extraction of entity mentions and relations. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Baltimore, MD, USA, 22–27 June 2014; Volume 1, pp. 402–412.
5. Ren, X.; Wu, Z.; He, W.; Qu, M.; Voss, C.R.; Ji, H.; Abdelzaher, T.F.; Han, J. Cotype: Joint extraction of typed entities and relations with knowledge bases. In Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 3–7 April 2017; pp. 1015–1024.
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. Yu, X.; Lam, W. Jointly identifying entities and extracting relations in encyclopedia text via a graphical model approach. In Proceedings of the 23rd International Conference on Computational Linguistics, Beijing, China, 23–27 August 2010; pp. 1399–1407.
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 (Volume 1: Long Papers), Melbourne, Australia, 15–20 July 2018; Volume 1, pp. 506–514.
9. Ye, H.; Zhang, N.; Deng, S.; Chen, M.; Tan, C.; Huang, F.; Chen, H. Contrastive Triple Extraction with Generative Transformer. In Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI, Virtually, 2–9 February 2021; pp. 14257–14265.
10. Fu, T.; Li, P.; Ma, W. Graphrel: Modeling text as relational graphs for joint entity and relation extraction. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 1409–1418.
11. Wei, Z.; Su, J.; Wang, Y.; Tian, Y.; Chang, Y. A novel cascade binary tagging framework for relational triple extraction. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 1476–1488.
12. Sun, K.; Zhang, R.; Mensah, S.; Mao, Y.; Liu, X. Progressive multitask learning with controlled information flow for joint entity and relation extraction. *Assoc. Adv. Artif. Intell.* **2021**, *35*, 13851–13859.
13. Brauwers, G.; Frasincar, F. A General Survey on Attention Mechanisms in Deep Learning. *IEEE Trans. Knowl. Data Eng.* **2021**. [CrossRef]
14. Lai, T.; Cheng, L.; Wang, D.; Ye, H.; Zhang, W. RMAN: Relational multi-head attention neural network for joint extraction of entities and relations. *Appl. Intell.* **2021**, *52*, 3132–3142. [CrossRef]
15. Zheng, S.; Wang, F.; Bao, H.; Hao, Y.; Zhou, P.; Xu, B. Joint extraction of entities and relations based on a novel tagging scheme. 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; Volume 1, pp. 1227–1236.
16. Peters, M.E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep contextualized word representations. *Proc. Conf. N. Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol.* **2018**, *1*, 2227–2237.
17. Devlin, J.; Chang, M.; Lee, K.; Toutanova, K. 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), Minneapolis, MN, USA, 2–7 January 2019; pp. 4171–4186.
18. Huang, W.; Cheng, X.; Wang, T.; Chu, W. BERT-Based Multi-head Selection for Joint Entity-Relation Extraction. In *Natural Language Processing and Chinese Computing. NLPCC 2019. Lecture Notes in Computer Science*; Tang, J., Kan, M.Y., Zhao, D., Li, S., Zan, H., Eds.; Springer: Berlin/Heidelberg, Germany, 2019; Volume 11839. [CrossRef]
19. Wadden, D.; Wennberg, U.; Luan, Y.; Hajishirzi, H. Entity, Relation, and Event Extraction with Contextualized Span Representations. 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), Hong Kong, China, 3–7 November 2019; Volume 2, pp. 5788–5793. [CrossRef]
20. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
21. Zhong, P.; Wang, D.; Miao, C. Knowledge-enriched transformer for emotion detection in textual conversations. 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), Hong Kong, China, 3–7 November 2019; Volume 1, pp. 165–176.

22. Gehring, J.; Auli, M.; Grangier, D.; Yarats, D.; Dauphin, Y.N. Convolutional sequence to sequence learning. In Proceedings of the 34th International Conference on Machine Learning (Proceedings of Machine Learning Research), Sydney, NSW, Australia, 6–11 August 2017; Volume 70, pp. 1243–1252.
23. Yu, F.; Koltun, V. Multi-Scale Context Aggregation by Dilated Convolutions. In Proceedings of the 4th International Conference on Learning Representations (ICLR), Caribe Hilton, San Juan, Puerto Rico, 2–4 May 2016; pp. 1–13.
24. Dauphin, Y.N.; Fan, A.; Auli, M.; Grangier, D. Language modeling with gated convolutional networks. In Proceedings of the 34th International Conference on Machine Learning, Sydney, NSW, Australia, 6–11 August 2017; Volume 70, pp. 933–941.
25. Luong, M.; Pham, H.; Manning, C.D. Effective approaches to attention-based neural machine translation. *arXiv* **2015**, arXiv:1508.04025.
26. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 July–1 August 2016; pp. 770–778. [CrossRef]
27. Ba, J.L.; Kiros, J.R.; Hinton, G.E. Layer normalization. *arXiv* **2016**, arXiv:1607.06450.
28. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2017**, arXiv:1412.6980.
29. Riedel, S.; Yao, L.; McCallum, A. Modeling relations and their mentions without labeled text. In Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Barcelona, Spain, 19–23 September 2010; pp. 148–163.
30. Gardent, C.; Shimorina, A.; Narayan, S.; Perez-Beltrachini, L. Creating training corpora for nlg micro-planners. 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. 179–188.
31. Zeng, X.; He, S.; Zeng, D.; Liu, K.; Liu, S.; Zhao, J. 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), Hong Kong, China, 3–7 November 2019; Volume 1, pp. 367–377.