electronics

MDPI

Article

Incorporating Entity Type-Aware and Word–Word Relation-Aware Attention in Generative Named Entity Recognition

Ying Mo and Zhoujun Li *

State Key Lab of Software Development Environment, Beihang University, Beijing 100191, China; moying@buaa.edu.cn
* Correspondence: lizj@buaa.edu.cn

Abstract: Named entity recognition (NER) is a critical subtask in natural language processing. It is particularly valuable to gain a deeper understanding of entity boundaries and entity types when addressing the NER problem. Most previous sequential labeling models are task-specific, while recent years have witnessed the rise of generative models due to the advantage of tackling NER tasks in the encoder–decoder framework. Despite achieving promising performance, our pilot studies demonstrate that existing generative models are ineffective at detecting entity boundaries and estimating entity types. In this paper, a multiple attention framework is proposed which introduces the attention of entity-type embedding and word–word relation into the named entity recognition task. To improve the accuracy of entity-type mapping, we adopt an external knowledge base to calculate the prior entity-type distributions and then incorporate the information input to the model via the encoder's self-attention. To enhance the contextual information, we take the entity types as part of the input. Our method obtains the other attention from the hidden states of entity types and utilizes it in self- and cross-attention mechanisms in the decoder. We transform the entity boundary information in the sequence into word–word relations and extract the corresponding embedding into the cross-attention mechanism. Through word–word relation information, the method can learn and understand more entity boundary information, thereby improving its entity recognition accuracy. We performed experiments on extensive NER benchmarks, including four flat and two long entity benchmarks. Our approach significantly improves or performs similarly to the best generative NER models. The experimental results demonstrate that our method can substantially enhance the capabilities of generative NER models.

Keywords: named entity recognition; attention; generative model

Citation: Mo, Y.; Li, Z. Incorporating Entity Type-Aware and Word–Word Relation-Aware Attention in Generative Named Entity Recognition. *Electronics* **2024**, *13*, 1407. https://doi.org/10.3390/electronics13071407

Academic Editor: Manohar Das

Received: 14 March 2024
Revised: 1 April 2024
Accepted: 4 April 2024
Published: 8 April 2024

1. Introduction

Named entity recognition (NER) is a subtask of information extraction that seeks to locate and classify named entities mentioned in unstructured text into predefined categories, such as the entity types of person, organization, and location. NER is one of the fundamental research problems in natural language processing, which has been widely adopted in information retrieval and question-answering systems [1–3]. Previous works [4–9] have addressed NER tasks with task-specific token-level sequential labeling or span-level classification methods. In token-level sequential labeling methods, each token is assigned a label to represent its entity type. On the other hand, span-level classification methods enumerate all possible spans in the sentences and classify them into predefined entity types.

Recently, sequence-to-sequence (seq2seq) generative approaches [10,11] have gained attention in the NER community due to their ability to jointly model all NER tasks in a unified framework. Although these models have shown promising results across all three NER categories (flat NER, nested NER, and discontinuous NER), NER faces two limitations that require the proposal of suitable methods for addressing them. First, these

generative models are ineffective at utilizing information about entity boundaries. For example, consider a seq2seq method tasked with identifying entities in a complex sentence; the model might successfully recognize "New York" as an entity but fail to discern the boundary between "New York" and "University" in the phrase "New York University". It incorrectly identifies the entire phrase as one entity instead of two separate entities: the location "New York", and the organization "University". This ineffectiveness stems from the autoregressive decoding process inherent in seq2seq models failing to capture the inter-word relationships within a sentence that are essential for identifying entity boundaries. Current seq2seq models generate coarse-grained entities and leverage context insufficiently, making a more precise and fine-grained approach required. Second, the seq2seq framework does not explicitly consider the effect of entity types on NER. While entity-type generation is based on the compounding tokens, the incorrect generation of these tokens from the decoder can lead to misguided entity-type mapping information. Access to prior knowledge from external entity databases aids the model in precisely determining entity types, and these entity types can in turn can shape the contextual learning of sentences, thereby improving the representation of entities. Therefore, it is crucial to incorporate entity-type mapping and entity boundary information into seq2seq NER models in order to overcome these limitations.

In this paper, we present a novel approach to address the limitations of existing seq2seq NER models. We propose a novel approach incorporating attention mechanisms built on the entity type and the word–word relation. Specifically, we leverage external knowledge bases such as Wikipedia to learn entity-type information, further improving entity-type mapping. We integrate entity-type distributions into the encoder. Within the decoder, we embed the entity type through self-attention and cross-attention mechanisms, improving the associative mapping between entities and their respective types. Furthermore, we integrate representations of word–word relations into the decoder in order to enhance the capability to distinguish entity boundaries.

Our proposed approach promotes the smooth incorporation of entity-type information and word pair relational knowledge into the seq2seq NER framework. The main contributions are summarized as follows:

- We propose a generative NER framework which merges entity-type embedding and word–word relation representation to improve named entity recognition performance.
- We leverage two novel attention mechanisms in the NER framework, namely, entity type-aware attention and word–word relation-aware attention, improving the interaction between entities, entity types, and word–word relations for better contextual information.
- We present a series of experiments demonstrating the effectiveness of our method against various baselines. Ablation studies further show the contribution of each component within our approach, confirming their individual effectiveness.

2. Related Work

2.1. Named Entity Recognition

Named entity recognition (NER) is a significant research area in natural language processing. Various methods have been proposed for named entity recognition. Traditional research on NER performed modeling as a sequence labeling task, primarily focusing on flat NER [4–6,12]. The focus later shifted to complex-structure NER [13–18], which is studied separately. Different approaches to NER include sequence labeling, span-based, hypergraph-based, and generative methods.

NER is typically treated as a sequence labeling problem which assigns a tag to each token and uses a sequence model [4,6,14,19–23] to predict labels of the sequences (e.g., BIO). Collobert et al. [20] introduce the linear-chain conditional random field (CRF) in convolutional neural networks (CNNs) and made the sequence labeling problem one of determining the respective likelihoods between adjacent tags. Strubell et al. [5] obtained features for the sentences via CNN. Following this work, Lample et al. [4] adopted bidirectional LSTM

with CRF to obtain the token representations. A number of studies [24,25] have combined CNNs and RNNs to extract features and learn word representations. Ju et al. [26] used dynamically stacking flat NER layers in the LSTM model for nested NER. Tang et al. [27] extended BIO to the BIOHD label scheme for discontinuous NER. However, these methods primarily require the design of different tagging schemes, and do not effectively address structurally complex or longer sentences such as nested and long NER. In contrast, our generative model can simultaneously handle this issue. Furthermore, we incorporate entity-type embedding from external knowledge and the encoder, while, word–word relation representation is merged to enhance entity boundary information.

Span-based methods commonly tackle complex NER tasks, e.g., nested NER. Researchers have proposed various approaches to obtain reasonable spans. Wang et al. [28] proposed a model which allows for interaction between spans from different layers. Yu et al. [15] utilized bi-affine attention to measure the possibility as a mention of span. Tan et al. [29] first predicted the boundary and then performed classification over the span features. Ouchi et al. [30] built a feature space with similar entity spans. However, these span-based models must enumerate all possible spans, while our model directly generates the entities. Li et al. [9] and Zhang et al. [31] presented a similar NER model, treating it as a span-based machine reading comprehension task. These methods require the design of template-based questions and multiple accesses to models, however, which raises computational costs; our proposed framework is able to avoid these issues. Lin et al. [32] offered a method that first detects the type of an anchor word and then locates the entity's boundaries. Shen et al. [16] presented a two-stage object detection method for nested entities. In addition to the above issues, these models have problems with gaps or chaining errors in the results of different stages. This paper proposes a method that enables interaction between entity type and entity boundary information to directly generate entity sequences.

Hypergraph-based methods have been proposed to cover many possible mentions in a sentence effectively. Lu et al. [33] introduced a method for joint mention extraction and classification using hypergraphs, followed by similar work from Muis et al. [23,34], who utilized a multigraph representation to address overlapping NER. Katiyar et al. [35] developed a hypergraph representation for nested entities, leveraging features extracted from an RNN. Wang et al. [7,36] proposed the idea of neural segmental hypergraphs. However, these models have trouble dealing with long inputs or many entity categories, as their hypergraph structures become extremely complex. These methods additionally struggle with spurious structure and structural ambiguity during inference [37]. In this paper, we aim to further investigate a simple and efficient model that learns entity type and boundary information for NER.

Generative methods treat the entity span sequence as a generation task, aiming to generate entities and entity types directly without requiring unique design of the tagging schema or ways to enumerate spans. Seq2seq methods have been proposed that directly generate entity label sequences from the input [5]. Strakova et al. [14] proposed a seq2seq method for nested NER that directly outputs the label of each token, associating the relation between words and labels via hard attention. Yan et al. [10] presented a pointer-based model which splices the label's embedding with the token representation. Zhang et al. [11] offered data augmentation from a causal perspective to generate entities. However, generative models do not fully utilize the entity boundary information implicit in the entity itself, which is critical for named entity recognition.

2.2. Attention Mechanism

Attention mechanisms have been employed in machine translation, machine comprehension, named entity recognition, and related natural language processing tasks. Attention-based methods have achieved impressive results, in that attention mechanisms have a large amount of memory and can build cooperation between the sequence of input. In addition, attention mechanisms allow models to automatically concentrate on the essential parts of the information while ignoring the less relevant details, thereby enhanc-

ing the model's ability to process complex data. The transformer model, first introduced by Vaswani et al. [38], has revolutionized the field of named entity recognition with its self-attention and cross-attention mechanisms, allowing long-range dependencies within the data to be captured. Based on this, many works [39,40] have leveraged attention mechanisms to integrate features from various sources of information, such as character representations, word embeddings, and position embedding. These integrated approaches enable models to focus selectively on the most pertinent information and select valuable knowledge, facilitating more nuanced language understanding. Ren et al. [39] proposed using an attention-based architecture over the word embedding and character-level component to learn the same semantic features for each word. Tan et al. [40] directly utilized attention mechanisms to capture the global dependencies of the input in order to enhance the performance of Chinese NER. TENER [41] is an adapted encoder based on attention mechanisms to merge the character- and word-level features. FLAT [42] uses a variant of self-attention to leverage the relative span position encoding.

3. Generative NER Task Formulation

In this section, we define the problem of seq2seq named entity recognition. Given an input sentence of n tokens $X = \{x_1, x_2, \ldots, x_n\}$, the goal of seq2seq NER is to generate a target sequence $Y = \{s_{11}, f_{11}, s_{12}, f_{12}, \ldots, s_{1k}, f_{1k}, g_1, \ldots s_{i1}, f_{i1}, s_{i2}, f_{i2}, \ldots, s_{ij}, f_{ij}, g_i\}$. Here, s and f are the starting and ending indices of a span, k is the span index in an entity, and $g_i \in \{g_1, \ldots, g_N\}$ is the entity type, where N is the total number of possible entity types. The generated schematic can be shown in Figure 1.

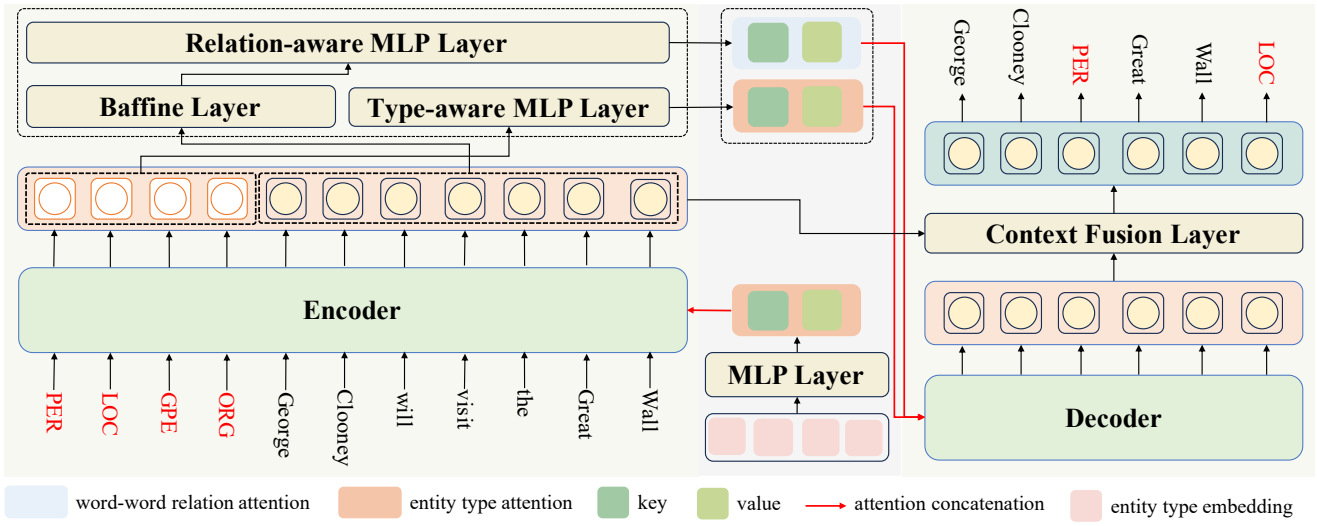

Figure 1. The architecture of our method, which contains both entity type-aware and word–word relation-aware attentions in the seq2seq framework. We incorporate the word–word relation-aware attention into the decoder and the entity type-aware attention into both the encoder and decoder. For the attention concatenation, refer to the process shown in the following sections.

4. Methodology

In this section, we introduce our proposed methodology. We outline the overall framework and describe the details of the modules (entity type-aware attention and word–word relation-aware attention) for implementing our method. The process in generative entities is introduced as well. Readers can expect a comprehensive overview of the supports that form the backbone of our framework.

4.1. Model Overview

Our model consists of an encoder that encodes the input sequence to its contextual embedding and a decoder that generates the output sequence with entity annotations. In

addition to the entity generation task, we design a relation representation learning task over the token pairs to better capture the correlations among the entities. The entity token relation attention and entity type attention are introduced and fused into the encoder and decoder as shown in Figure 1. We present the details of each component separately in the following subsections.

4.2. Entity Type Aware Attention

Heterogeneous factors such as entity types and entity boundaries [15,43–45] greatly impact named entity recognition. In this section, we discuss the modeling of entity types in our seq2seq NER framework, allowing for interactions with the input sequences and guiding the model to learn more effective token representation. We merge the entity type-aware attention in the encoder and the decoder.

In the encoder, we incorporate entity types as part of the input, which are concatenated with the given sentence. To achieve better representations of entities, we try to obtain the entity type aware attention from an external entity base. Prior knowledge can be obtained to improve the NER task. Then, we feed the entity type-aware attention into the self-attention layers in the encoder.

As shown in Figure 2, the comprehensive representations of entity types are obtained from an external entity base incorporated into the encoder through entity type attention.

The representation of an entity type T is a weighted sum of the entity representations. For example, assuming that the entity label set is $G = \{person, location, organization\}$, as shown in Figure 2, if entity type $T = location$ contains entity set $C_T = \{Beijing, London, \ldots, New\ York\}$, the initial embedding of entity type T can be obtained as follows:

$$E_T = \theta_i E_{C_T^1} + \cdots + \theta_i E_{C_T^i} \tag{1}$$

where $E_{C_T^i} \in \mathbb{R}^{d_h}$ is the embedding of the i-th entity in the entity set C_T and θ_i is the weight of the entity, such as its frequency. Note that if the entity types are not in the external source, we may obtain entity type embeddings by random initialization or entity type tokens.

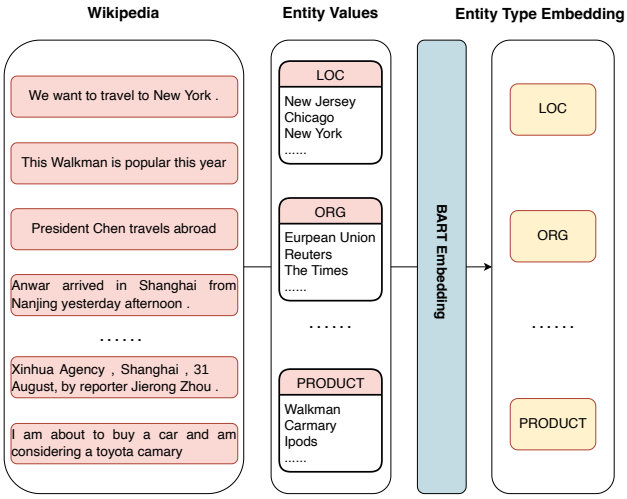

Figure 2. Entity type embedding from external entity knowledge.

To leverage the entity type information, we design an entity type-aware attention and integrate it into the corresponding self-attention layers. The vectors $K_E \in \mathbb{R}^{B \times N_h \times N \times d_k}$ and $V_E \in \mathbb{R}^{B \times N_h \times N \times d_k}$ represent the key and value of the entity type-aware attention, which are concatenated with the original key $K \in \mathbb{R}^{B \times N_h \times n \times d_k}$ and value $V \in \mathbb{R}^{B \times N_h \times n \times d_k}$ vectors in the encoder as follows:

$$R^*((K_E, V_E)) = \mathrm{MLP}_{type-aware}(E) \tag{2}$$

$$head^l = Attention(Q^l, (K_E^l \oplus K^l), (V_E^l \oplus V^l)) \tag{3}$$

where B is the batch size, N_h denotes the number of the heads, $head^l$ is the head representation of the l-th layer, K_E, V_E are obtained from the embedding E by an MLP layer [46], $E \in \mathbb{R}^{N \times d_h}$ is the embedding of entity types with N as the number of entity types, $Q \in \mathbb{Q}^{B \times N_h \times n \times d_k}$ denotes the query of the attention, and $\oplus$ means the concatenation. Figure 3 shows the merging of the entity type attention. The entity type-aware attention from the external entity knowledge base is only applied to self-attention layers in the encoder.

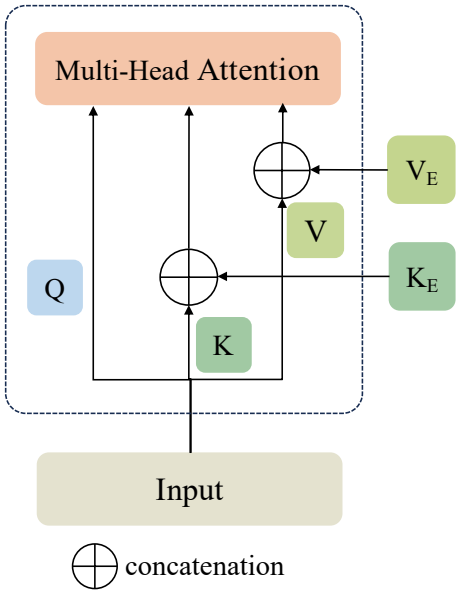

Figure 3. Concatenation in a self-attention layer. In the encoder, K_E and V_E are calculated from the entity type embedding of external entity knowledge via the type-aware MLP layer. In the decoder, K_E and V_E are calculated from the hidden states of entity type tokens in the encoder via the type-aware MLP layer.

In the decoder, E is from the hidden states of the entity type tokens, which are part of the input. Then, we use Equations (2) and (3) to apply the entity type-aware attention. In this process, we apply it to all self-attention and cross-attention layers.

4.3. Word–Word Relation-Aware Attention

Improving entity boundary detection is crucial for named entity recognition; therefore, we integrate information relevant to entity boundaries into our framework. We utilize word–word relation representations as feature information to learn about entity boundaries. Specifically, inspired by [45,47–49], which can be used to enhance relations between tokens in an entity for NER, we extract these word–word relation features to improve the representation of the predicted token in the decoder.

Given a sentence $X = \{x_1, \ldots, x_n\}$, we obtain the hidden representation $H = \{h_1 \ldots, h_n\}$ after the encoder layers. The word–word relation representation set $R\{r_{ij} | (i, j \in [1, n])\} \in \mathbb{R}^{n \times n \times d_h}$ in the sentence r_{ij} is the relation of word pair (x_i, x_j), and is obtained through the bi-affine layer.

$$\begin{aligned} s_i &= \mathrm{MLP}(h_i), e_j = \mathrm{MLP}(h_j) \\ r_{ij} &= s_i^\top W_1 e_j + W_2(s_i \oplus e_j) + b, \end{aligned} \tag{4}$$

where W_1, W_2, and b denote the trainable parameters and MLP is the fully connected layer.

The word–word relation representations r_{ij} reflect the entity boundary information and the syntactic structure of the sentence. These representations are valuable knowledge with which the model can perceive and learn entity boundaries more accurately, which plays a vital role in identifying entities. As the above r_{ij} cannot be directly used in our proposed word–word relation-aware attention mechanism, we feed the representations

between words into an MLP layer to obtain useful features. Meanwhile, this process can generate keys and values that match the attention mechanism inherent in the seq2seq model itself. Finally, the obtained word–word relation-aware attention key $K_R \in \mathbb{R}^{B\times N_h\times M\times d_k}$ and value $V_R \in \mathbb{R}^{B\times N_h\times M\times d_k}$ matrices are concatenated with the original key $K \in \mathbb{K}^{B\times N_h\times n\times d_k}$ and value $V \in \mathbb{R}^{B\times N_h\times n\times d_k}$, respectively, in the cross-attention layers to enhance the model generation with regard to the entity boundaries. The entity relation-aware attention is defined as follows:

$$R^*((K_R, V_R)) = \text{MLP}_{relation-aware}(R) \tag{5}$$

$$head^l = Attention(Q^l, (K_R^l \oplus K_E^l \oplus K^l), (V_R^l \oplus V_E^l \oplus V^l)) \tag{6}$$

where B is the batch size, N_h denotes the number of heads, M is the length of the word–word relation-aware attention using the convolution operation, K_R^l and V_R^l are from K_R and V_R, respectively, and K_E^l and V_E^l are calculated from the entity type embedding described in the above section. However, the entity type embedding is the hidden state of the entity types as a part of the input sequence. Figure 4 shows the word–word relation-aware attention incorporated into the cross-attention layer.

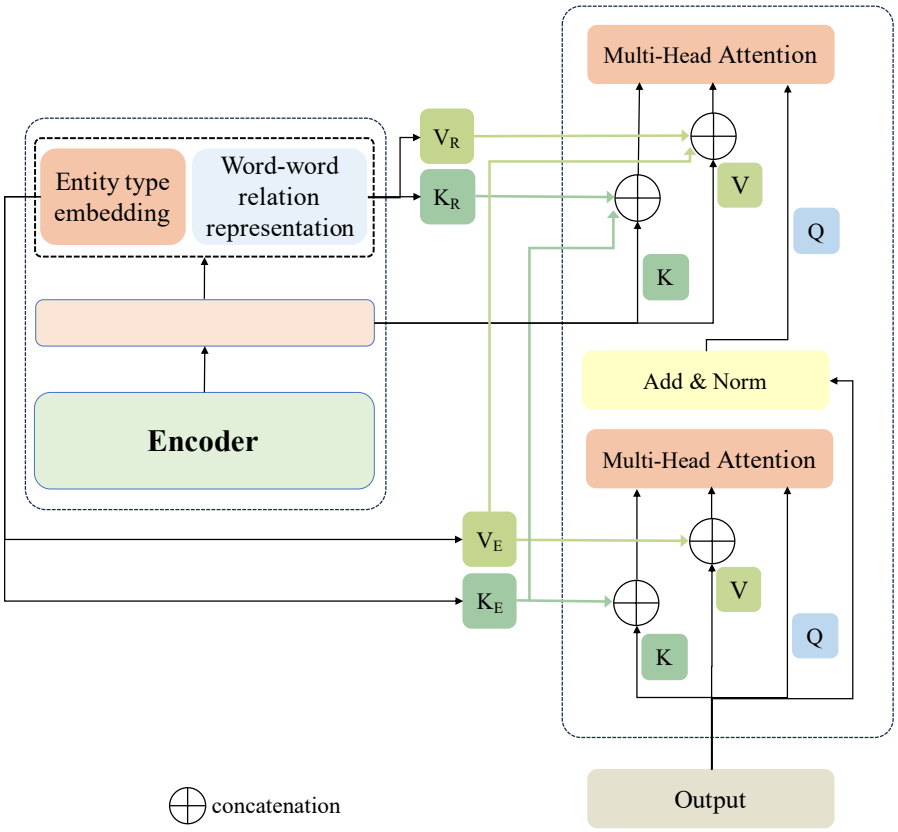

Figure 4. Concatenation in a self-attention and cross-attention layer. In the decoder, K_E and V_E are calculated from the hidden states of entity type tokens in the encoder by the type-aware MLP layer., while K_R and V_R are calculated from the word–word relation representations by the bi-affine and relation-aware MLP layer.

4.4. Entity Decoding

The decoder decodes the token embedding from the encoder to generate the entities. In particular, at step t, the decoder acquires the token embedding $h_t^d \in \mathbb{R}^{d_h}$ based on the encoder output and all the previous decoded tokens as follows:

$$h_t^d = Decoder(H^e, Y_{<t}^{\wedge}) \tag{7}$$

where $H^e \in \mathbb{R}^{n \times d_h}$, $Y^{\wedge}_{<t} = \left[y^{\wedge}_1, \ldots, y^{\wedge}_{t-1}\right]$ is the generated token sequence before t. To enhance the accuracy of the generated tokens, we introduce a context fusion layer to further decode the output. We join the entity type representations and hidden state of the tokens in the encoder to the respective hidden states of the output, which can help to improve the representation of context within the sentence:

$$\bar{h_t}^d = (H^e \otimes h^d_t) \oplus (E \otimes h^d_t) \tag{8}$$

where E is the entity type representation in the encoder and $\otimes$ denotes the dot product.

Finally, the output token index distribution P_t can be obtained by the function

$$P_t = Softmax(\bar{h_t}^d). \tag{9}$$

For a given sequence $X = \{x_1, x_2, \ldots, x_n\}$, we attempt to minimize the negative log-likelihood concerning the corresponding ground truth labels, which can be defined as

$$L = -\log p(Y|X). \tag{10}$$

5. Experiments

This experiments section provides our detailed experimental setup, including the NER datasets evaluated, the backbones with parameter settings, the evaluation metrics, and the baseline models for comparison. In addition, we discuss the main results of the datasets compared to the baselines. This section aims to provide readers with insight into the validation of the proposed method and its performance against the baselines.

5.1. Datasets

We evaluated NER on CoNLL 2003 [12] and OntoNotes 5.0 (https://catalog.ldc.upenn.edu/LDC2013T19 (accessed on 12 November 2023)) [50] in English, following the settings splits in prior works [4,15,51]. We performed further testing using OntoNotes 4.0 (https://catalog.ldc.upenn.edu/LDC2011T03 (accessed on 12 November 2023)) [52], MSRA [53], Weibo [54,55], and Resume [56] in Chinese.

To further validate the effectiveness of the model, we evaluated the long NER datasets EBM-NLP (https://github.com/bepnye/EBM-NLP/) (accessed on 12 November 2023) [57] and SemEval 2017 (https://scienceie.github.io/resources.html (accessed on 12 November 2023)) [58] as well. EBM-NLP annotates PICO spans, defining the Participants, Interventions, Comparisons, and Outcomes in a clinical trial paper [59]. We processed it following the works [60–62]. SemEval 2017 is a task involving extracting keyphrases and relations from documents with mention-level keyphrase identification (the types are PROCESS, TASK, and MATERIAL), mention-level keyphrase classification, and mention-level semantic relation extraction between keyphrases. We merged the first two subtasks as the NER task and prepared it following [58]. The statistics of the datasets are listed in Tables 1 and 2.

Table 1. Statistics of NER datasets other than SemEval 2017 and EBM-NLP.

	CoNLL 2003	Ontonotes 5.0	Ontonotes 4.0	MSRA	Resume	Weibo
#sentences	20,744	76,714	24,371	48,442	4759	1890
#entities	36,431	111,868	30,783	81,249	16,565	2677

Table 2. Statistics of the SemEval 2017 and EBM-NLP datasets.

	SemEval 2017	EBM-NLP
#entities	5730	63,693
#unique entities	1697	47,916
#single-word entities	18%	19%
#entities, word length ≥ 3	51%	51%
#entities, word length ≥ 5	22%	30%

5.2. Implementation Details

We adopted BART-Large as the backbone network. Following previous works [10,45], the encoder and decoder had twelve layers with 1024 dimensional embedding for our experiments. For the English datasets, we used the BART-Large model [63]. For the Chinese datasets, we used the BART-Large-Chinese model [64]. We used the AdamW [65] optimizer. We executed a grid search of the hyperparameters shown in Table 3 and selected the set of parameters that had the best performance on the validation set. The batch size was 32 for OntoNotes 5.0 and 16 for the others. Finally, we used 1×10^{-5} for the BART-Large model and 5×10^{-5} for the other components. When deriving the key K_E and value V_E of the self-attention from the entity type embedding, we employed multilayer MLP similar to the proj_down-proj_up structure [66], with down dim 512 and up dim 1024. For most baseline methods, the hyperparameters were set according to the experimental configurations in the original papers. However, there were a few variations; for the SemEval 2017 and EBM-NLP datasets, the batch size and max sequence length were the same as those used for our proposed method.

Table 3. Hyperparameters used to train our model.

Hyper-Parameters	Range	Final
Batch Size	[16, 32, 64]	16/32
Dropout	[0.2, 0.3, 0.5]	0.5
Learning Rate for Bart	[5×10^{-6} , 1×10^{-5}, 5×10^{-5}, 1×10^{-4}]	1×10^{-5}
Learning Rate for Other components	[5×10^{-6}, 1×10^{-5}, 5×10^{-5}, 1×10^{-4}]	5×10^{-5}
Weight Decay	[0, 0.01]	0.01
Warmup Ratio	[0, 0.01]	0.01
Max Sequence Length	[64, 128, 512]	128
Down_dim	[128, 256, 512]	512
Up_dim	[512, 768, 1024]	1024
MLP-hidden size	[128, 256]	128

5.3. Evaluation Metrics

An entity was considered to be correctly predicted if the entity label and boundary matched the ground truth. Following prior works [10,11,33], we computed the precision (P), recall (R), and F1 (F) scores for each dataset, utilizing the F1 score on the validation set as the criterion for selecting the optimal model. We ran each experiment five times and report the average metrics.

5.4. Comparative Baselines

We compared our method with several previous baselines:

- LSTM-CRF/Stack-LSTM [4] and ID-CNNs [5] offer iterated dilated convolutional neural networks.
- SH [7] propose the use of hypergraphs to address the NER task.
- Seq2Seq [14] obtains tokens with the labels in a sequence.
- BiaffineNER [15] offers a bi-affine module for span-based models to explore spans.
- BartNER [10] offers a unified NER model with a pointer generating the start–end indexes of entities and types.

- DebiasNER [11] designs data augmentation to eliminate incorrect biases from a causality perspective.
- W2NER [37] models the unified NER as word–word relation classification to tackle the different NER tasks.
- LatticeLSTM [56] probes a lattice LSTM encoding the characters and words matching a lexicon.
- TENER [41] uses an encoder to consider character, word, direction, relative distance, and unscaled attention.
- LGN [67] uses a lexicon-based graph neural network with global semantics to interact among characters, words, and sentence semantics.
- FLAT [42] is a flat-lattice model that converts the lattice structure into a flat structure.
- Lexicon [68] uses lexical knowledge in Chinese NER based on a collaborative graph network.
- LR-CNN [69] uses a CNN-based approach with lexicons via a rethinking mechanism.
- PLTE [70] uses the characters and matches lexical words in parallel via the transformer.
- SoftLexcion [71] merges the word lexicon into the character representations and adjusts the character representation layer.
- MECT [72] uses a multi-metadata embedding-based cross-transformer that fuses the characters' structural information.
- SciBERT [60], PubMedBERT [61], and VarMAE [62] are pretrained models based on BERT [73] focusing on the scientific and biomedical domains. Finally, TIAL_UM [58] ranks first on the SemEval 2017 keyphrase extraction leaderboard.

5.5. Results

We compared our model with seq2seq models [10,11,14], sequence labeling [4,5], span-based methods [15], hypergraph models [7], and more.

The results on all the datasets are shown in Tables 4–6. Several key observations can be made from the comparison results. Our model performs better than some sequence labeling, span-based, and generative models on most datasets. For CoNLL2003, our method outperforms sequence labeling [4], the span-based method [15], and the hypergraph-based method [7] by 2.6%, 1.02% and 3.04% in terms of F1 score. Compared with the seq2seq models (Seq2Seq [14], BartNER [10], and DebiasNER [11]), CoNLL 2003 shows increases of 0.56%, 1.02%, and 0.4%. For OntoNotes 5.0, our method improves the F1 score by 4.0% and 0.34% compared with the ID-CNNs [5] and W2NER [37] baselines. Compared with generative models BartNER and DebiasNER, there is an improvement of 0.46% and 0.42%, respectively, in terms of the F1 score. For the Chinese NER datasets, all demonstrate improvements in performance to varying extents. Compared to the generative method BartNER, our method shows increases of 5.99%, 0.76%, 2.75%, and 7.68%, respectively. For SemEval 2017 and EBM-NLP, the performance of our method shows significant improvement compared with both BartNER and W2NER. These experimental results validate our hypothesis that our approach can effectively model word–word relations, thereby improving entity decoding. Moreover, the entity type information incorporated into our method further boosts model performance.

The results of our approach show slight improvement overall compared to W2NER. The reason for this is that W2NER unites the position region-aware representation of the grid and the relation representation of token pairs to estimate entities. Although our model uses word–word relation representation, it has only coarse granularity and no directionality during inference.

Table 4. Results on the CoNLL2003 and OntoNotes 5.0 datasets; results are statistically significant with p-value < 0.005. The best scores are in bold, while the second-best scores are underlined.

Model	CoNLL2003			OntoNotes 5.0		
	P	R	F	P	R	F
LSTM-CRF/Stack-LSTM [4]	-	-	90.94	-	-	-
ID-CNNs [5]	-	-	90.65	-	-	86.84
SH [7]	-	-	90.50	-	-	-
Seq2Seq [14]	-	-	92.98	-	-	-
BiaffineNER [15]	92.91	92.13	92.52	90.01	89.77	89.89
BartNER [10]	92.61	**93.87**	93.24	89.99	90.77	90.38
DebiasNER [11]	92.78	93.51	93.14	89.77	91.07	90.42
W2NER [37]	92.71	93.44	93.07	90.03	90.97	90.50
Ours	**93.14**	93.65	**93.54**	**90.13**	**91.59**	**90.84**

Table 5. Results on the Chinese NER datasets; results are statistically significant with p-value < 0.005. The best scores are in bold, while the second-best scores are underlined.

MODEL	OntoNotes 4.0			MSRA			Resume			Weibo		
	P	R	F	P	R	F	P	R	F	P	R	F
LatticeLSTM [56]	76.35	71.56	73.88	93.57	92.79	93.18	94.81	94.11	94.46	53.04	62.25	58.79
TENER [41]	-	-	72.43	-	-	92.74	-	-	95.00	-	-	58.17
LGN [67]	76.40	72.60	74.45	94.50	92.93	93.71	95.37	94.84	95.11	57.14	66.67	59.92
FLAT [42]	-	-	81.82	-	-	96.09	-	-	95.86	-	-	68.55
Lexicon [68]	75.06	74.52	74.79	94.01	92.93	93.47	-	-	-	-	-	63.09
LR-CNN [69]	76.40	72.60	74.45	94.50	92.93	93.71	95.37	94.84	95.11	-	-	59.92
PLTE[BERT] [70]	79.62	81.82	80.60	94.91	94.15	94.53	96.16	96.75	96.45	**72.00**	66.67	69.23
SoftLexicon [71]	**83.41**	82.21	82.81	95.75	95.10	95.42	96.08	96.13	96.11	70.94	67.02	70.50
MECT[BERT] [72]	-	-	82.57	-	-	96.24	-	-	95.98	-	-	70.43
BartNER [10]	79.18	80.21	79.69	95.30	94.75	95.86	96.72	90.90	94.21	67.25	63.88	65.52
W2NER [37]	82.31	83.36	83.08	96.12	**96.08**	96.10	**96.96**	96.35	96.65	70.84	**73.87**	72.32
Ours	83.39	**87.92**	**85.68**	**96.26**	95.93	**96.62**	96.03	**96.85**	**96.96**	69.87	68.32	**72.68**

Table 6. Results on the Long NER SemEval 2017 and EBM-NLP datasets; results are statistically significant with p-value < 0.005. The best scores are in bold, while the second-best scores are underlined.

Model	SemEval 2017			EBM-NLP		
	P	R	F	P	R	F
SciBERT [60]	-	-	-	-	-	71.18
PubMedBERT [61]	-	-	-	-	-	**73.38**
VarMAE [62]	-	-	-	-	-	76.01
TIAL_UW [58]	-	-	44.00	-	-	-
BartNER [10]	38.36	47.92	42.16	51.22	47.84	40.96
W2NER [37]	49.92	44.68	47.16	**66.22**	38.84	48.96
Ours	**53.69**	**63.83**	**57.64**	63.16	**68.61**	71.89

It can be seen that the improvement of our model over the baselines on the EBM-NLP datasets is marginal compared to PubMedBERT [61]. The primary reason for this is that PubMedBERT is a domain-specific pre-trained language model utilizing biomedical data. Nonetheless, our model is capable of attaining comparable or superior performance.

6. Analysis and Discussion

In this analysis and discussion section, we analyze the results obtained from the various components within our method to show the performance of our proposed approach. Then, to assess our method's robust ability to process long sentences and long entities, we

explore the implications of our approach and compare its performance with that of the baselines. Finally, by analyzing some instances of existing incorrect predictions, we aim to provide insights for potential future work. This section contextualizes our results, and we expect that the analysis will benefit future work.

6.1. Ablation Study

To estimate the impact of various components within our method, we conducted ablation studies by sequentially omitting each component, namely, word–word relation-aware attention (rel-att) and entity type-aware attention (type-att). In this section, we designate the seq2seq model devoid of "rel-att" and "type-att" as the "baseline". We refer to the model that includes word–word relation-aware attention as "+ rel-att". The same nomenclature applies to "+ type-att" and "+ rel-att & type-att".

The outcomes of the ablation studies are presented in Tables 7–9. The results indicate that our proposed method significantly outperforms the baseline when incorporating word–word relation attention (+ rel-att) or entity type attention (+ type-att). This validates the effectiveness of both word–word relation attention and entity type attention. Moreover, integrating word–word relation attention and entity type-aware attention into an NER framework yields the most promising results.

Table 7. Ablation studies on the CoNLL2003 and OntoNotes 5.0 datasets.

Model	CoNLL2003	OntoNotes 5.0
baseline	92.82	90.02
+ rel-att	93.19 ↑**0.37**	90.32 ↑**0.30**
+ type-att	93.39 ↑**0.57**	90.48 ↑**0.46**
+ rel-att & type-att	93.54 ↑**0.72**	90.84 ↑**0.82**

Table 8. Ablation studies on the Chinese NER datasets.

Model	OntoNotes 4.0	MSRA	Resume	Weibo
baseline	82.73	95.01	94.56	65.32
+ rel-att	83.97 ↑**1.24**	95.77 ↑**0.76**	95.73 ↑**1.17**	68.55 ↑**3.23**
+ type-att	84.81 ↑**2.08**	95.98 ↑**0.97**	96.34 ↑**1.78**	69.46 ↑**4.14**
+ rel-att & type-att	85.68 ↑**2.95**	96.62 ↑**1.61**	96.96 ↑**2.40**	72.68 ↑**7.36**

Table 9. Ablation studies on the Long NER SemEval 2017 and EBM-NLP datasets.

Model	SemEval 2017	EBM-NLP
baseline	52.55	66.15
+ rel-att	55.69 ↑**3.14**	69.61 ↑**3.46**
+ type-att	56.75 ↑**4.20**	70.24 ↑**4.09**
+ rel-att & type-att	57.64 ↑**5.09**	71.89 ↑**5.74**

The improvements have subtle differences considering the differences in the entity types, domains, and complexity of entities. For the Chinese NER datasets, the improvement is relatively significant. In addition to their simple structures, the external entity type embeddings mitigate the diverse and complex expressions of Chinese entities, leading to a closer representation of entities within the same type. For example, the entity types in Weibo have similarities, such as per.nam and per.nom, representing specific and general persons, respectively (i.e., "张三" is a per.nam, and "男人" is labeled as per.nom). Integrating external entities to construct the embedding of entity types can help to enhance the distinction between them. Furthermore, including the entity type as part of the input enhances the in-context learning within the sentence. Leveraging entity type-aware attention during decoding further reinforces the mapping between entity textual information and entity types. Additionally, given Chinese tokenization traits, introducing

word pair relation representation helps to understand the entity boundary information. On the SemEval 2017 and EBM-NLP datasets, the performance of our proposed components is better compared with the baseline. Our method can generate various length sequences and learn more information from the entity type and word–word relation attention.

6.2. Effect on Long Sentences

In this section, we split the test set by sentence length in order to assess our method's ability to process long sentences. The results are shown in Figure 5. Specifically, we categorized the test sets into subsets based on sentence length (#words), with ranges set at [0–10, 10–20, 30–40, 50–60, ≥60]. It is obvious that our method provides the largest overall gain compared to the generative model on long sentences (≥60 words) [10]. Note that the amount of data with sentence lengths in [50–60, ≥60] is relatively tiny compared to other groups; thus, the evaluation is relatively high. Nonetheless, our model performs better than the baseline models, particularly in long-sentence scenarios.

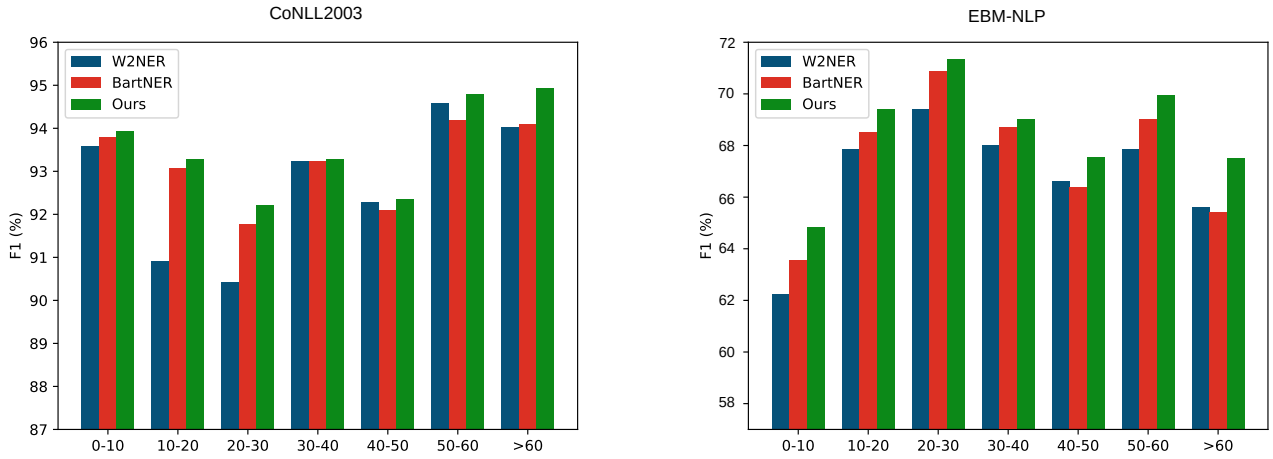

Figure 5. Results when sentence length changes (#words).

6.3. Effect on Long Entities

To verify the ability of our method to handle long entities, we report the experimental results in Figure 6. In this section, we selected BartNER and W2NER as baselines. Considering the number of long entities, we set the entity length in the range from 1 to 5 for CoNLL2003, 1 to 6 for SemEval 2017, and 1 to 8 for EBM-NLP. As can be observed, there is little change in the performance of the sets with a small length of entities. However, on CoNLL2003 with long entities (E(L) $\geq$ 5), the F1 score improves by up to 3%. For SemEval 2017, when the entity length is less than 5, the effect difference of each model is not noticeable; however, when it is greater than 5, the effect of our model is nearly 7% higher than the other models [10,37]. For EBM-NLP, our method provides the largest gain compared to the other models on long entities (E(L) $\geq$ 8).

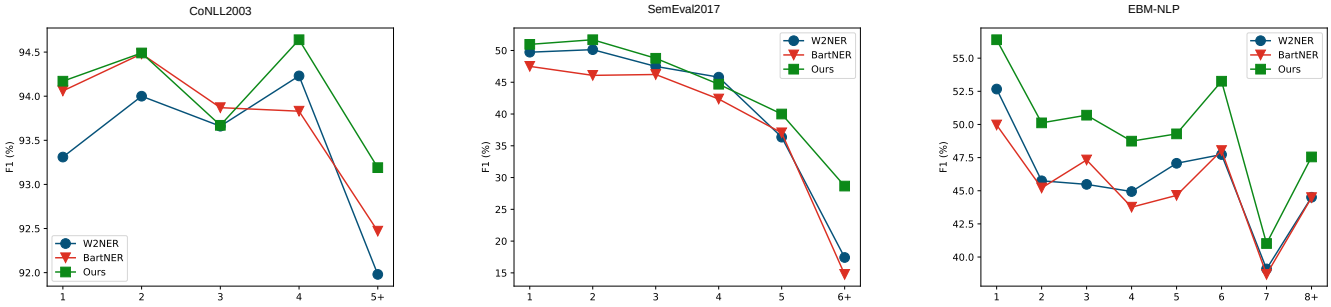

Figure 6. Results on various datasets when entity length changes.

6.4. Effectiveness for Entity Boundary

We ran experiments to analyze the effectiveness of entity boundary recognition, e.g., for Instance #1 in Table 10, the prediction "(中韩, gpe)" has an error in entity boundary detection even though the generated entity type is "gpe". In this analysis, we only consider the entity boundary metric and overlook whether the entity type is correct. The F1 scores of different models on the datasets are shown in Figure 7. It can be observed that our method has a positive impact on entity boundary detection. For CoNLL2003 and OntoNotes 4.0, our method performs slightly better than W2NER, which employs relative position representation and fine-grained token pair relation representation. For SemEval 2017, our proposed approach is significantly superior to the other models. These results are consistent with our expectation that leveraging the entity type-aware attention and word–word relation-aware attention into the generative NER framework can contribute to enhancing entity boundary detection performance.

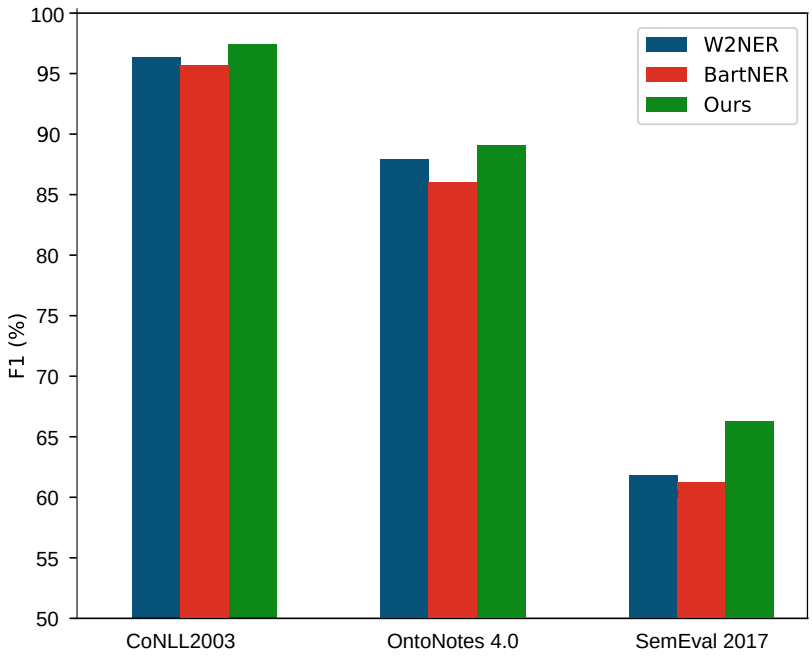

Figure 7. Results on entity boundary detection.

Table 10. Error analysis. Text in color indicates predicted entities that are incorrect.

Instance
instance #1: 中韩经贸研讨会即将在北京举行 **Pred:** (中韩, GPE), (北京, GPE) **Gold:** (中, GPE), (韩, GPE), (北京, GPE)
instance #2 : the Office of Fair Trade called for British Airways/American to allow third-party access to their joint frequent flyer programme where the applicant does not have access to an equivalent programme . **Pred:** (Office of Fair Trade, ORG), (British Airways American, ORG) **Gold:** (Office of Fair Trade, ORG), (British Airways American, ORG)
instance #3: These data were converted to standard triangulation language (STL) surface data as an aggregation of fine triangular meshes using 3D visualization and measurement software (Amira version X , FEI , Burlington , MA , USA). **Pred:** (standard triangulation language, Process), (triangular meshes, Material), (3D visualization, Process) **Gold:** (standard triangulation language, Process), (triangular meshes, Material), (3D visualization, Process), (Amira version X, Material)
instance #4: South Africa's trip to Kanpur for the third test against India has given former England test cricketer Bob Woolmer the chance of a sentimental return to his birthplace . **Pred:** (South Africa, LOC), (Kanpur, LOC), (India, LOC), (England, LOC), (Bob Woolmer, PER) (test cricketer, PER) **Gold:** (South Africa, LOC), (Kanpur, LOC), (India, LOC), (England, LOC), (Bob Woolmer, PER)

6.5. Case Study

We selected a number of instances for analysis to promote further future works in the field of NER. We have classified the incorrect entities into four classes, as shown in Table 10.

Incorrect Boundaries: As instance #1 shows, the generated entity has an incorrect boundary. This sentence of instance #1 in Chinese mentions two countries, which were misidentified as a single entity. In Chinese, the entity words denote abbreviations of the two countries, and the model did not learn this knowledge, which conveys the boundary difficulty of the entity. For multiple nested entities in sentences, the model is more prone to misjudgment. Therefore, enhancing the learning of entity boundary information can improve model performance.

Distract Context: As instance #2 shows, our model predicts the incorrect entity type because of the ambiguous contexts that may be expressed in a similar context or lacking the descriptive context. The same tokens may have different meanings in various semantic contexts. For accurate recognition of entity types, it is necessary for the model to learn a comprehensive understanding of sentence context and entity types in order to make the correct judgment.

Missing Entities: As instance #3 shows, the result misses an entity. This may be because the entity is rare or specific, caused by the unbalanced learning which makes the model tend to judge sentences with a similar context to high-frequency entities. For this type of error, improving the model's understanding of critical information in sentences can be achieved by enhancing the attention mechanisms and attempting data augmentation.

Extra Entities: As instance #4 shows, our model predicts extra entities that look right but are not in the gold set. The reason for this may be that the entity appears repeatedly in other sentences or the data are noisy. Combination with other tasks, such as entity boundary detection, entity linking, and entity disambiguation, can helpto prevent excessive entity recognition.

7. Conclusions

In this work, we have introduced a novel approach that merges entity type and word–word relation into the generative NER framework to achieve better performance.

Specifically, we combine entity type and word–word relation by attention mechanisms with the original attention in the backbone network, improving the model's ability to discriminate entity types and detect entity boundaries. We further take the entity types as special tokens and as part of the input for learning valuable knowledge from the context. Experiments on various benchmarks show the superior performance of our method. Integrating entity types as special tokens further enriches the model's context learning, allowing for more precise entity recognition. Furthermore, introducing entity type attention further strengthens the connection between entity tokens in the sentence and predefined entity types. We transform the information of the entity boundary to the relations of word pairs and merge it in the proposed framework via the attention mechanism, including the syntactic and semantic relationships between words to enhance the accuracy of entity boundary detection. However, there are weaknesses in the potential increase of model complexity and evaluation time. To address these issues, we will explore further approaches, such as non-autoregressive methods, which can speed up the decoding. Because our proposed method primarily relies on a large amount of annotated data, we will focus on generative NER models based on large language models in a low-resource setting and consider how to integrate resources such as images. The practical implications of our proposed approach extend to relation extraction, where accurate entity recognition is crucial. We are trying to use this method to improve knowledge graph construction.

Author Contributions: Y.M.: conceptualization, methodology, formal analysis, software, writing—original draft. Z.L.: supervision, writing—review and editing. 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 (Grant Nos. 62276017, U1636211, 61672081) and the Fund of the State Key Laboratory of Software Development Environment (Grant No. SKLSDE-2021ZX-18).

Data Availability Statement: MDPI Research Data Policies at https://github.com/weizhepei/CasRel, (accessed on 15 September 2020).

Conflicts of Interest: The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Aliod, D.M.; van Zaanen, M.; Smith, D. Named Entity Recognition for Question Answering. In *Proceedings of the ALTA 2006*; Cavedon, L., Zukerman, I., Eds.; Australasian Language Technology Association: Sydney, NSW, Australia, 2006; pp. 51–58.
2. Li, Q.; Ji, H. Incremental Joint Extraction of Entity Mentions and Relations. In Proceedings of the ACL 2014, Baltimore, MD, USA, 22–27 June 2014; pp. 402–412.
3. Zhong, Z.; Chen, D. A Frustratingly Easy Approach for Entity and Relation Extraction. In Proceedings of the NAACL-HLT 2021, Online, 6–11 June 2021; pp. 50–61.
4. Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; Dyer, C. Neural Architectures for Named Entity Recognition. In Proceedings of the NAACL HLT 2016, San Diego, CA, USA, 12–17 June 2016; pp. 260–270.
5. Strubell, E.; Verga, P.; Belanger, D.; McCallum, A. Fast and Accurate Entity Recognition with Iterated Dilated Convolutions. In Proceedings of the EMNLP 2017, Copenhagen, Denmark, 9–11 September 2017; pp. 2670–2680.
6. Panchendrarajan, R.; Amaresan, A. Bidirectional LSTM-CRF for Named Entity Recognition. In Proceedings of the PACLIC 2018, Hong Kong, China, 1–3 December 2018; Politzer-Ahles, S., Hsu, Y., Huang, C., Yao, Y., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA , 2018.
7. Wang, B.; Lu, W. Combining Spans into Entities: A Neural Two-Stage Approach for Recognizing Discontiguous Entities. In Proceedings of the EMNLP, Hong Kong, China, 3–7 November 2019; pp. 6215–6223.
8. Yu, B.; Zhang, Z.; Sheng, J.; Liu, T.; Wang, Y.; Wang, Y.; Wang, B. Semi-Open Information Extraction. In Proceedings of the WWW 2021, Online, 19–23 April 2021; pp. 1661–1672.
9. Li, X.; Feng, J.; Meng, Y.; Han, Q.; Wu, F.; Li, J. A Unified MRC Framework for Named Entity Recognition. In Proceedings of the ACL 2020, Virtual, 5–10 July 2020; pp. 5849–5859.
10. Yan, H.; Gui, T.; Dai, J.; Guo, Q.; Zhang, Z.; Qiu, X. A Unified Generative Framework for Various NER Subtasks. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*; Association for Computational Linguistics: Kerrville, TX, USA, 2021; pp. 5808–5822.
11. Zhang, S.; Shen, Y.; Tan, Z.; Wu, Y.; Lu, W. De-Bias for Generative Extraction in Unified NER Task. In Proceedings of the ACL 2022, Dublin, Ireland, 22–27 May 2022; pp. 808–818.

12. Sang, E.F.T.K.; Meulder, F.D. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In Proceedings of the Seventh Conference on Natural Language Learning, CoNLL 2003, Held in cooperation with HLT-NAACL 2003, Edmonton, AB, Canada, 31 May– 1 June 2003; pp. 142–147.
13. Karimi, S.; Metke-Jimenez, A.; Kemp, M.; Wang, C. Cadec: A corpus of adverse drug event annotations. *J. Biomed. Inform.* **2015**, *55*, 73–81. [CrossRef] [PubMed]
14. Straková, J.; Straka, M.; Hajic, J. Neural Architectures for Nested NER through Linearization. In Proceedings of the ACL 2019, Florence, Italy, 28 July–2 August 2019; pp. 5326–5331.
15. Yu, J.; Bohnet, B.; Poesio, M. Named Entity Recognition as Dependency Parsing. In Proceedings of the ACL 2020, Virtual, 5–10 July 2020; pp. 6470–6476.
16. Shen, Y.; Ma, X.; Tan, Z.; Zhang, S.; Wang, W.; Lu, W. Locate and Label: A Two-stage Identifier for Nested Named Entity Recognition. In Proceedings of the ACL/IJCNLP, Bangkok, Thailand, 1–6 August 2021; pp. 2782–2794.
17. Dai, X.; Karimi, S.; Hachey, B.; Paris, C. An Effective Transition-based Model for Discontinuous NER. In Proceedings of the ACL, 2020, Virtual, 5–10 July 2020; pp. 5860–5870.
18. Li, F.; Lin, Z.; Zhang, M.; Ji, D. A Span-Based Model for Joint Overlapped and Discontinuous Named Entity Recognition. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*; Association for Computational Linguistics: Kerrville, TX, USA, 2021; pp. 4814–4828.
19. Ratinov, L.; Roth, D. Design Challenges and Misconceptions in Named Entity Recognition. In Proceedings of the Thirteenth Conference on Computational Natural Language Learning, CoNLL 2009, Boulder, CO, USA, 4–5 June 2009; pp. 147–155.
20. Collobert, R.; Weston, J.; Bottou, L.; Karlen, M.; Kavukcuoglu, K.; Kuksa, P.P. Natural Language Processing (Almost) from Scratch. *J. Mach. Learn. Res.* **2011**, *12*, 2493–2537.
21. Metke-Jimenez, A.; Karimi, S. Concept Identification and Normalisation for Adverse Drug Event Discovery in Medical Forums. In Proceedings of the BMDID@ISWC, Kobe, Japan, 17 October 2016.
22. Chiu, J.P.C.; Nichols, E. Named Entity Recognition with Bidirectional LSTM-CNNs. *Trans. Assoc. Comput. Linguist.* **2016**, *4*, 357–370. [CrossRef]
23. Muis, A.O.; Lu, W. Learning to Recognize Discontiguous Entities. In Proceedings of the EMNLP 2016, Austin, TX, USA, 1–5 November 2016; pp. 75–84.
24. Ma, X.; Hovy, E.H. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. In Proceedings of the ACL 2016, Berlin, Germany, 7–12 August 2016.
25. Zhou, P.; Zheng, S.; Xu, J.; Qi, Z.; Bao, H.; Xu, B. Joint Extraction of Multiple Relations and Entities by Using a Hybrid Neural Network. In *Proceedings of the Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data—16th China National Conference, CCL 2017, and 5th International Symposium, NLP-NABD 2017*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2017 ; Volume 10565, pp. 135–146.
26. Ju, M.; Miwa, M.; Ananiadou, S. A Neural Layered Model for Nested Named Entity Recognition. In *Proceedings of the NAACL-HLT 2018*; Walker, M.A., Ji, H., Stent, A., Eds.; Association for Computational Linguistics: Kerrville, TX, USA, 2018; pp. 1446–1459.
27. Tang, B.; Hu, J.; Wang, X.; Chen, Q. Recognizing continuous and discontinuous adverse drug reaction mentions from social media using LSTM-CRF. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 2379208. [CrossRef]
28. Wang, J.; Shou, L.; Chen, K.; Chen, G. Pyramid: A Layered Model for Nested Named Entity Recognition. In Proceedings of the ACL 2020, Virtual, 5–10 July 2020; pp. 5918–5928.
29. Tan, C.; Qiu, W.; Chen, M.; Wang, R.; Huang, F. Boundary Enhanced Neural Span Classification for Nested Named Entity Recognition. In Proceedings of the AAAI 2020, New York, NY, USA, 7–12 February 2020; pp. 9016–9023.
30. Ouchi, H.; Suzuki, J.; Kobayashi, S.; Yokoi, S.; Kuribayashi, T.; Konno, R.; Inui, K. Instance-Based Learning of Span Representations: A Case Study through Named Entity Recognition. In Proceedings of the ACL 2020, Virtual, 5–10 July 2020; pp. 6452–6459.
31. Zhang, F.; Ma, L.; Wang, J.; Cheng, J. An MRC and adaptive positive-unlabeled learning framework for incompletely labeled named entity recognition. *Int. J. Intell. Syst.* **2022**, *37*, 9580–9597. [CrossRef]
32. Lin, H.; Lu, Y.; Han, X.; Sun, L. Sequence-to-Nuggets: Nested Entity Mention Detection via Anchor-Region Networks. In Proceedings of the ACL 2019, Florence, Italy, 28 July–2 August 2019; pp. 5182–5192.
33. Lu, W.; Roth, D. Joint mention extraction and classification with mention hypergraphs. In Proceedings of the EMNLP 2015, Lisbon, Portugal, 17–21 September 2015; pp. 857–867.
34. Muis, A.O.; Lu, W. Labeling Gaps Between Words: Recognizing Overlapping Mentions with Mention Separators. In Proceedings of the EMNLP 2017, Copenhagen, Denmark, 9–11 September 2017; pp. 2608–2618.
35. Katiyar, A.; Cardie, C. Nested Named Entity Recognition Revisited. In Proceedings of the NAACL-HLT 2018, New Orleans, LO, USA, 1–6 June 2018; pp. 861–871.
36. Wang, Y.; Yu, B.; Zhu, H.; Liu, T.; Yu, N.; Sun, L. Discontinuous Named Entity Recognition as Maximal Clique Discovery. In Proceedings of the ACL/IJCNLP 2021, Bangkok, Thailand, 1–6 August 2021; pp. 764–774.
37. Li, J.; Fei, H.; Liu, J.; Wu, S.; Zhang, M.; Teng, C.; Ji, D.; Li, F. Unified Named Entity Recognition as Word-Word Relation Classification. In Proceedings of the AAAI 2022, Virtual, 22 February–1 March 2022; pp. 10965–10973.

38. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All you Need. In Proceedings of the Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
39. Rei, M.; Crichton, G.K.O.; Pyysalo, S. Attending to Characters in Neural Sequence Labeling Models. In Proceedings of the COLING 2016, Osaka, Japan, 11–16 December 2016; pp. 309–318.
40. Tan, Z.; Wang, M.; Xie, J.; Chen, Y.; Shi, X. Deep Semantic Role Labeling With Self-Attention. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th Innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, LO, USA, 2–7 February 2018; pp. 4929–4936.
41. Yan, H.; Deng, B.; Li, X.; Qiu, X. TENER: Adapting Transformer Encoder for Named Entity Recognition. *arXiv* **2019**, arXiv:1911.04474.
42. Li, X.; Yan, H.; Qiu, X.; Huang, X. FLAT: Chinese NER Using Flat-Lattice Transformer. In Proceedings of the ACL 2020, Virtual, 5–10 July 2020; pp. 6836–6842.
43. Fu, Y.; Tan, C.; Chen, M.; Huang, S.; Huang, F. Nested Named Entity Recognition with Partially-Observed TreeCRFs. *AAAI Conf. Artif. Intell.* **2021**, *35*, 12839–12847. [CrossRef]
44. Aly, R.; Vlachos, A.; McDonald, R. Leveraging Type Descriptions for Zero-shot Named Entity Recognition and Classification. In Proceedings of the ACL/IJCNLP 2021, Bangkok, Thailand, 1–6 August 2021; pp. 1516–1528.
45. Mo, Y.; Tang, H.; Liu, J.; Wang, Q.; Xu, Z.; Wang, J.; Wu, W.; Li, Z. Multi-Task Transformer with Relation-Attention and Type-Attention for Named Entity Recognition. In Proceedings of the ICASSP 2023, Rhodes Island, Greece, 4–10 June 2023; pp. 1–5.
46. Liu, R.; Li, Y.; Tao, L.; Liang, D.; Zheng, H.T. Are we ready for a new paradigm shift? A survey on visual deep mlp. *Patterns* **2022**, *3*, 100520. [CrossRef]
47. Mo, Y.; Yang, J.; Liu, J.; Wang, Q.; Chen, R.; Wang, J.; Li, Z. mCL-NER: Cross-Lingual Named Entity Recognition via Multi-view Contrastive Learning. *arXiv* **2023**, arXiv:2308.09073.
48. Shang, Y.; Huang, H.; Mao, X. OneRel: Joint Entity and Relation Extraction with One Module in One Step. In Proceedings of the AAAI 2022, Virtual, 22 February–1 March 2022; pp. 11285–11293.
49. Zhu, E.; Li, J. Boundary Smoothing for Named Entity Recognition. In Proceedings of the ACL 2022, Dublin, Ireland, 22–27 May 2022; pp. 7096–7108.
50. Pradhan, S.; Moschitti, A.; Xue, N.; Ng, H.T.; Björkelund, A.; Uryupina, O.; Zhang, Y.; Zhong, Z. Towards Robust Linguistic Analysis using OntoNotes. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning, CoNLL 2013, Sofia, Bulgaria, 8–9 August 2013; pp. 143–152.
51. Pradhan, S.; Moschitti, A.; Xue, N.; Uryupina, O.; Zhang, Y. CoNLL-2012 Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes. In Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning—Proceedings of the Shared Task: Modeling Multilingual Unrestricted Conference in OntoNotes, EMNLP-CoNLL 2012, Jeju Island, Republic of Korea, 12–14 July 2012; pp. 1–40.
52. Weischedel, R.; Pradhan, S.; Ramshaw, L.; Palmer, M.; Xue, N.; Marcus, M.; Taylor, A.; Greenberg, C.; Hovy, E.; Belvin, R.; et al. *Ontonotes Release 4.0. LDC2011T03*; Linguistic Data Consortium: Philadelphia, PA, USA, 2011.
53. Levow, G.A. The third international Chinese language processing bakeoff: Word segmentation and named entity recognition. In Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing, Sydney, Australia, 22–23 July 2006; pp. 108–117.
54. He, H.; Sun, X. F-Score Driven Max Margin Neural Network for Named Entity Recognition in Chinese Social Media. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, 3–7 April 2017; Volume 2: Short Papers, 2017; Association for Computational Linguistics: Kerrville, TX, USA, 2017; pp. 713–718.
55. Peng, N.; Dredze, M. Named entity recognition for chinese social media with jointly trained embeddings. In Proceedings of the EMNLP 2015, Lisbon, Portugal, 17–21 September 2015; pp. 548–554.
56. Zhang, Y.; Yang, J. Chinese NER Using Lattice LSTM. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, 15–20 July 2018; Volume 1: Long Papers; Association for Computational Linguistics: Kerrville, TX, USA, 2018; pp. 1554–1564.
57. Nye, B.E.; Li, J.J.; Patel, R.; Yang, Y.; Marshall, I.J.; Nenkova, A.; Wallace, B.C. A Corpus with Multi-Level Annotations of Patients, Interventions and Outcomes to Support Language Processing for Medical Literature. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, 15–20 July 2018; Volume 1: Long Papers; Association for Computational Linguistics: Kerrville, TX, USA, 2018; pp. 197–207.
58. Augenstein, I.; Das, M.; Riedel, S.; Vikraman, L.; McCallum, A. SemEval 2017 Task 10: ScienceIE - Extracting Keyphrases and Relations from Scientific Publications. In Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, BC, Canada, 3–4 August 2017; pp. 546–555.
59. Kim, S.; Martínez, D.; Cavedon, L.; Yencken, L. Automatic classification of sentences to support Evidence Based Medicine. *BMC Bioinform.* **2011**, *12*, S5. [CrossRef]
60. Beltagy, I.; Lo, K.; Cohan, A. SciBERT: A Pretrained Language Model for Scientific Text. In Proceedings of the EMNLP-IJCNLP 2019, Hong Kong, China, 3–7 November 2019; pp. 3613–3618.

61. Gu, Y.; Tinn, R.; Cheng, H.; Lucas, M.; Usuyama, N.; Liu, X.; Naumann, T.; Gao, J.; Poon, H. Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. *ACM Trans. Comput. Health* **2022**, *3*, 2:1–2:23. [CrossRef]
62. Hu, D.; Hou, X.; Du, X.; Zhou, M.; Jiang, L.; Mo, Y.; Shi, X. VarMAE: Pre-training of Variational Masked Autoencoder for Domain-adaptive Language Understanding. In Proceedings of the EMNLP Findings 2022, Abu Dhabi, United Arab Emirates, 7–11 December 2022; pp. 6276–6286.
63. Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; Zettlemoyer, L. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. *arXiv* **2019**, arXiv:1910.13461.
64. Shao, Y.; Geng, Z.; Liu, Y.; Dai, J.; Yang, F.; Zhe, L.; Bao, H.; Qiu, X. CPT: A Pre-Trained Unbalanced Transformer for Both Chinese Language Understanding and Generation. *arXiv* **2021**, arXiv:2109.05729.
65. Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. In Proceedings of the ICLR 2019, New Orleans, LA, USA, 6–9 May 2019.
66. Li, X.L.; Liang, P. Prefix-Tuning: Optimizing Continuous Prompts for Generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*; Association for Computational Linguistics: Kerrville, TX, USA, 2021; pp. 4582–4597.
67. Gui, T.; Zou, Y.; Zhang, Q.; Peng, M.; Fu, J.; Wei, Z.; Huang, X. A Lexicon-Based Graph Neural Network for Chinese NER. In Proceedings of the EMNLP-IJCNLP 2019, Hong Kong, China, 3–7 November 2019; pp. 1040–1050.
68. Sui, D.; Chen, Y.; Liu, K.; Zhao, J.; Liu, S. Leverage Lexical Knowledge for Chinese Named Entity Recognition via Collaborative Graph Network. In Proceedings of the EMNLP-IJCNLP 2019, Hong Kong, China, 3–7 November 2019; pp. 3828–3838.
69. Gui, T.; Ma, R.; Zhang, Q.; Zhao, L.; Jiang, Y.; Huang, X. CNN-Based Chinese NER with Lexicon Rethinking. In Proceedings of the IJCAI 2019, Macao, China, 10–16 August 2019; pp. 4982–4988.
70. Xue, M.; Yu, B.; Liu, T.; Zhang, Y.; Meng, E.; Wang, B. Porous Lattice Transformer Encoder for Chinese NER. In Proceedings of the COLING 2020, Online, 8–13 December 2020; pp. 3831–3841.
71. Ma, R.; Peng, M.; Zhang, Q.; Wei, Z.; Huang, X. Simplify the Usage of Lexicon in Chinese NER. In Proceedings of the ACL 2020, Virtual, 5–10 July 2020; pp. 5951–5960.
72. Wu, S.; Song, X.; Feng, Z. MECT: Multi-Metadata Embedding based Cross-Transformer for Chinese Named Entity Recognition. In Proceedings of the ACL/IJCNLP 2021, Bangkok, Thailand, 1–6 August 2021; Association for Computational Linguistics: Kerrville, TX, USA, 2021; pp. 1529–1539.
73. Devlin, J.; Chang, M.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv1810.04805.