future internet

MDPI

Article

An Improved Method for Named Entity Recognition and Its Application to CEMR†

Ming Gao, Qifeng Xiao, Shaochun Wu * and Kun Deng

Department of Intelligent Information Processing, Shanghai University, Shanghai 200444, China

* Correspondence: scwu@shu.edu.cn

† This paper is an extended version of our paper published in This paper is an extended version of our paper: Ming Gao, Qifeng Xiao, Shaochun Wu, Kun Deng, An attention-based ID-CNNs-CRF model for named entity recognition on clinical electronic medical records. In the Proceedings of the 28th International Conference on Artificial Neural Networks, 17th–19th September, 2019, Munich, Germany; No.195.

Received: 8 July 2019; Accepted: 15 August 2019; Published: 26 August 2019

Abstract: Named Entity Recognition (NER) on Clinical Electronic Medical Records (CEMR) is a fundamental step in extracting disease knowledge by identifying specific entity terms such as diseases, symptoms, etc. However, the state-of-the-art NER methods based on Long Short-Term Memory (LSTM) fail to exploit GPU parallelism fully under the massive medical records. Although a novel NER method based on Iterated Dilated CNNs (ID-CNNs) can accelerate network computing, it tends to ignore the word-order feature and semantic information of the current word. In order to enhance the performance of ID-CNNs-based models on NER tasks, an attention-based ID-CNNs-CRF model, which combines the word-order feature and local context, is proposed. Firstly, position embedding is utilized to fuse word-order information. Secondly, the ID-CNNs architecture is used to extract global semantic information rapidly. Simultaneously, the attention mechanism is employed to pay attention to the local context. Finally, we apply the CRF to obtain the optimal tag sequence. Experiments conducted on two CEMR datasets show that our model outperforms traditional ones. The F1-scores of 94.55% and 91.17% are obtained respectively on these two datasets, and both are better than LSTM-based models.

Keywords: clinical electronic records; named entity recognition; convolutional neural network

1. Introduction

With the rapid development of the medical industry, data mining on Clinical Electronic Medical Records (CEMR) plays an important role in precision medicine. It provides basic technology for subsequent medical record summaries, computer auxiliary diagnosis, etc. The state-of-the-art Named Entity Recognition (NER) methods use Long Short-Term Memory (LSTM) to extract features and then employ the Conditional Random Field (CRF) to obtain the optimal tag sequence [1–4]. However, the temporal structure of LSTM-based models is usually computationally expensive and inefficient, especially when faced with massive medical records. The output of the current time of the LSTM model depends on the output of the previous moment, which results in the inability to calculate in parallel. The CNN model tends to have a faster prediction time than the LSTM model of the temporal structure. People often have to make a choice between the excellent performance and low efficiency of the LSTM model on the NLP task [5,6]. This would be very beneficial if the CNN model directly replaces the LSTM structure and achieves similar performance to the LSTM on NLP tasks. Recent works [7,8] attempted to apply Convolutional Neural Networks (CNN) to NER. Most of the deep learning networks are based on the convolutional neural network (CNN) architecture because it can make better use of the GPU and thus have a great performance in terms of speed. Nevertheless, the performance of CNNs-based models is poorer than LSTMs-based models due to it neglecting the

global semantic information. Recently, the Iterated Dilated CNNs (ID-CNNs) [9] were proposed to efficiently aggregate a broad context. The comparison between the IDCNN model and the traditional Bi-LSTM model structure can be seen in Figure 1. However, this model ignores the significance of the word-order feature and local context in the text. As shown in Figure 1, the natural timing structure of the traditional Bi-LSTM model enables location information to be captured. The ID-CNNs model has no time structure. It cannot capture the relative relationship between words and ignores word-order information. For example, swapping the positions of x_1 and x_2 in Figure 1 has a large impact on the Bi-LSTM model. For the ID-CNN model, the output is not affected.

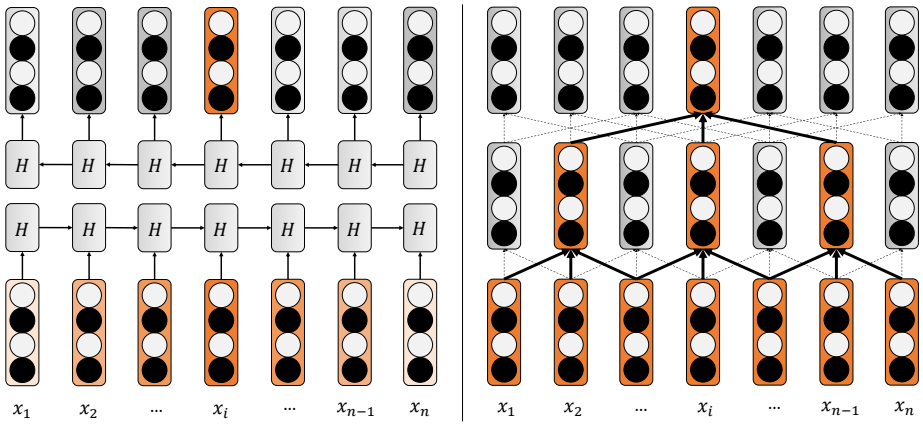

Figure 1. The left figure shows the traditional Bi-LSTM model. The input of the current moment of the Bi-LSTM model depends on the output of the previous moment, so at time i, the calculation must be iterative from left to right and iteratively calculated from right to left. The figure on the right is the Iterated Dilated (ID)-CNNs model. It is similar to the CNN structure and can be calculated in parallel. In addition, the receptive field is expanded between layers, so that global semantic information can be obtained.

To address these issues, we propose an attention-based ID-CNNs-CRF model. We first introduce position embedding to capture word-order information. Then, the attention mechanism is applied to the ID-CNNs-CRF model because of its good performance in NLP tasks, which enables the enhancement of the influence of critical words. Finally, we apply the CRF to obtain the optimal tag sequence. Experimental results on two CEMR datasets demonstrate that the proposed model achieves better prediction performance with a higher F1-score than those of baseline methods.

The remainder of the paper is structured as follows: Section 2 introduces the related work. Section 3 describes the details of the proposed method. Section 4 demonstrates the proposed methodology with a series of experiments. Finally, the conclusions of this study are given in Section 5.

2. Related Work

Benefiting from the implementation of digital medicine, the digitization of medical records has been promoted for many years. Named Entity Recognition (NER) of CEMR designed to identify critical entities of interest is a basic step in medical information extraction. Traditionally, many simple, but straightforward methods, such as rule-based and heuristic-search-based methods, have been utilized to identify critical medical entities [10]. These methods tend to achieve a low recall value due to the inability to cover all medical entities. Although rule-based methods seem better than dictionary-based methods, large numbers of rules require extensive domain knowledge to be formulated by medical professionals [11]. With the expansion of medical data, these time-consuming and laborious original methods seem to be clumsy. However, these approaches still exist because they can be utilized as part of other systems to achieve good performance [12,13].

NER methods based on statistical machine learning include Support Vector Machine (SVM) [14], the Hidden Markov Model (HMM) [15,16], the Maximum Entropy Hidden Markov

Model (MEHMM) [17], and the Conditional Random Field (CRF) [18]. Zhou et al. [16] presented an HMM NER system to deal with the special phenomena in the biomedical domain. Suwias et al. [19] utilized a CRF-based machine learning system named Nersuite to achieve an F-score of 88.46% on the Gellus corpus. The CRF methods have decent performance, but rely heavily on the selection of features. At present, the CRF-based medical named entity recognition method is the best method in statistical machine learning because of the consideration of the transfer between tags. These methods do not need to match a large number of medical entity dictionaries, nor do experts need to make rules for entity boundaries, but rely heavily on feature selection. These features such as Parts-Of-Speech (POS), lexical features, capitalization, etc., need linguists and domain experts to formulate them, which means that feature engineering is required.

In recent years, deep learning methods have been developed for NER. LSTM architectures that are capable of learning the long-term dependencies have been put forward [20], especially bidirectional recurrent neural network architectures [20–22]. Lample et al. [1] utilized bidirectional LSTMs and conditional random fields to obtain improvement in NER in multiple languages without resorting to any language-specific knowledge. Marc-Antoine et al. [3] utilized NeuroCRF to obtain an F1-score of 89.28% on the WikiNER dataset. Ling et al. proposed a neural network approach named attention-based Bi-LSTM-CRF for document-level chemical NER. The approach leverages document-level global information obtained by the attention mechanism to enforce tagging consistency across multiple instances of the same token in a document [23]. However, they are inefficient because of their sequential processing on sentences. To alleviate this problem, Emma et al. [7] applied the ID-CNNs architecture to speed up the processing of the network. They proved that the test-time speed of the ID-CNN models was 1.42-times faster than the Bi-LSTM models. However, these models tend to ignore the word-order feature and local context compared to LSTM-based models. As shown in Figure 1, the ID-CNN model failed to take advantage of word-order information between words, so the model itself is more like a bag of words model. More importantly, due to the stacking structure of ID-CNN, the output of the last layer has gained too much receptive field, so there is a lack of perception of the local environment. On the one hand, we need to make the ID-CNN model fuse location information. On the other hand, more context information about the current location of the model should be considered when predicting the output of the current location, rather than global information.

In this paper, in order to promote the performance of the ID-CNNs-CRF model, position embedding is utilized to introduce word-order information, and the attention mechanism is applied to focus on those critical words by assigning different weights.

3. Method

In this section, we construct a variety of features as the model input and then describe the proposed attention-based ID-CNNs-CRF model in detail.

3.1. Pretreatment

3.1.1. Data Formatting

It is difficult to perform many preprocessing operations on the original text for NER tasks, especially stop words, special symbols, etc., because many medical record entities may contain connectors and other special characters. In preprocessing, we do half angle conversion and traditional character conversion to ensure that the model vocabulary is moderate. In addition, we utilize the uniform symbol NUM to represent Numbers, ENG to represent English words, and UNK to represent Unknown words.

3.1.2. Tagging Scheme

NER is a type of sequence tagging task that assigns a label to each entity that contains multiple tokens. The IOB2 annotation [24] is the tagging scheme mainly utilized in sequence tagging to express an entity. It uses "B", which means "Begin", to indicate the beginning of the entity, "I", which means "Inside", to indicate the non-beginning part of the entity, and "O", which means "Other", to indicate the non-entity. These IOB tags are added in front of the entity class as labels for tokens. For example, B-Symptom means the begging of a Symptom entity.

3.2. The Construction of Features

In order to get richer semantic features, we constructed a module for extracting word representation. Word representation is a concatenation of word embedding, character embedding, POS embedding, and position embedding. The extraction module of word representation is shown in Figure 2.

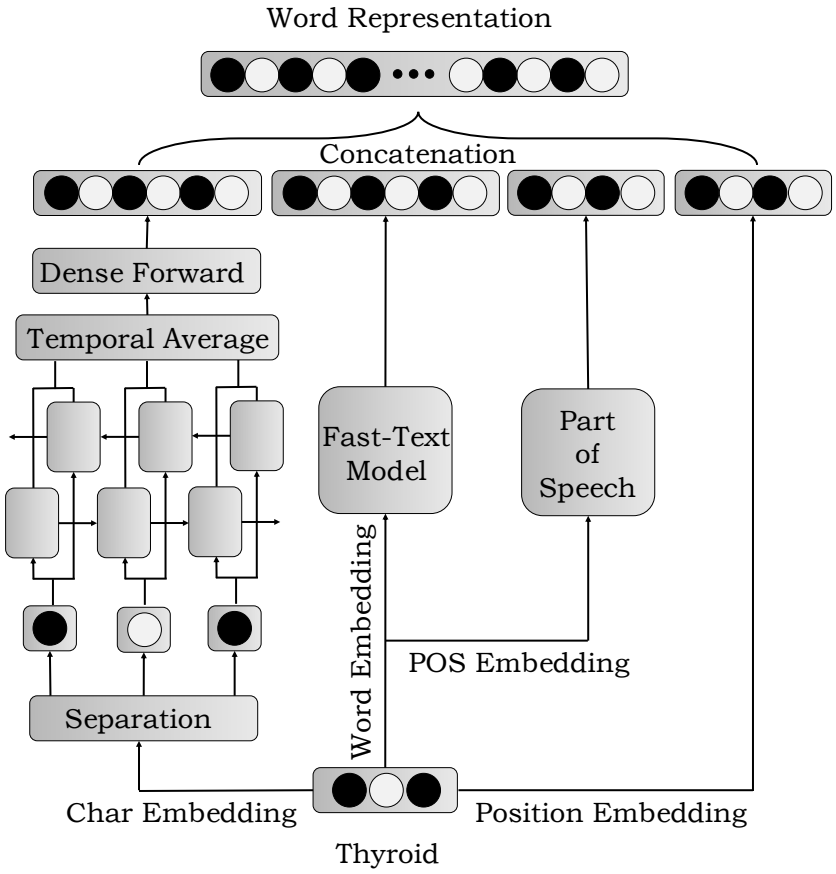

Figure 2. The extraction module of the word representation. The one on the far left is character embedding; the middle one is word embedding; and the two on the right are part of speech embedding and position embedding.

3.2.1. Word Embedding

Distributed representations such as Word2vec [25] and Fast-Text [26] are becoming the preferred choice for word vector representation because they can get an abstract representation of the word. We chose Fast-Text [26] instead of Word2vec [27] to characterize words because it is not only fast in massive text, but also relieves the out-of-vocabulary problem. In order to obtain high-quality word

characterization, a large amount of medical record data was crawled from the Internet [28] as a training corpus of the word embedding. We preprocessed the corpus and used Jieba [29] to segment the words, then utilized Fast-Text to get the word embedding.

3.2.2. Char Embedding

The quality of word embedding is affected by word segmentation, which makes the model introduce errors of word segmentation from the beginning. To obtain a high-quality word representation and avoid word segmentation errors, we took the character of the word as a sequence and then obtained the character level representation through the bidirectional recurrent neural network. For the output of the bidirectional cyclic neural network, we took the temporal average as the final output.

3.2.3. POS Embedding

The performance of NLP tasks relies heavily on word representation, especially linguistic features. We utilized Jieba [29] (a Chinese word segmentation tool) to segment each sentence and obtained the parts-of-speech (such as nouns, verbs, etc). We obtained the POS representation of each part-of-speech by random embedding and set them to be trainable for fine-tuning.

3.2.4. Position Embedding

Different from the recurrent neural networks, the ID-CNN network architecture does not have a natural word-order structure, so it cannot capture word-order information. To obtain the position information, we numbered each position (each position corresponds to a number) and then introduced a certain position embedding for each word by random embedding.

3.3. Attention-Based ID-CNNs-CRF Model

Firstly, we obtained word representation according to the description in Section 3.2 and then obtained the output at each position by utilizing iterated dilated CNNs. Secondly, the attention mechanism was employed to pay attention to the local context. Finally, we used CRF to learn the rules of transfer among labels. Figure 3 illustrates the proposed attention-based ID-CNNs-CRF model.

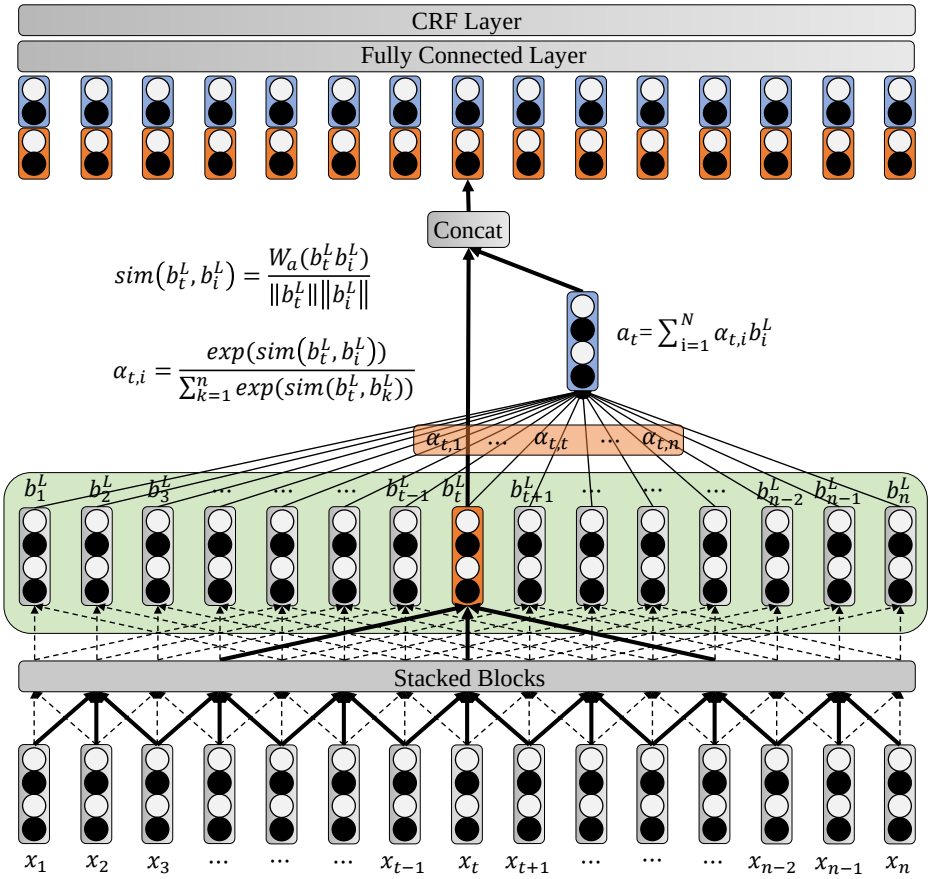

Figure 3. Attention-based ID-CNN-CRF architecture. We stacked 4 Dilated CNN blocks, each as shown in Figure 5. To simplify the drawing, we used *Stacked Blocks* to represent all the layers in the middle. For the output b_t^L at position t (the orange unit with four circles), we calculated its similarity with all the output units and took them as weights $\{\alpha_{t,1}, ..., \alpha_{t,i}, ..., \alpha_{t,n}\}$. The output was multiplied by the corresponding weight and then summed as the attention vector a_t. Concatenate a_t with b_t^L, and add a full connection layer to map it into the category space. Finally, the optimal tag sequence was obtained by CRF.

3.3.1. Iterated Dilated CNNs

The Dilated CNN model proposed by Fisher et al. [30] is different from the normal CNN filter. Normal CNN acts on a continuous position of the input matrix and continuously slides to do convolution and pooling. The dilated CNN model adds a dilation width to the filter. When it acts on the input matrix, it skips the input data in the middle of all the dilation width, and the size of the filter matrix itself remains the same, so the filter achieves a wider input, as is shown in Figure 4.

For the input sequence $X = \{x_1, x_2, ..., x_t, ..., x_n\}$, we denote the j-th dilated convolutional layer of dilation with width δ as D_δ^j. We applied a dilation-1 convolution D_1^0 to transform the input as a representation i_t and took it as the beginning of the stack layers.

$$c_t^{(0)} = i_t = D_1^{(0)} x_t \tag{1}$$

Next, the stacked layers are represented as follows:

$$c_t^{(j)} = r(D_{2^{L_c-1}}^{(j-1)} c_t^{(j-1)}) \tag{2}$$

where $c_t^{(j)}$ denotes the output of the j-th dilated convolutional layer of dilation, L_c denotes the layers of dilated convolutions, and r denotes the activation function of the ReLU activation function.

Each block had three layers of expansion convolution (excluding input), so the output of each block was $c_t^{(3)}$. Figure 5 illustrates the architecture of a block. Our model stacks four blocks of the same architectures, and each is a three-layer dilated convolution layer with the dilation width of $[1, 1, 2]$. We define these three layers of expansion convolution as a block B.

$$b_t^{(1)} = B(x_t) \tag{3}$$

$$b_t^{(k)} = B(b_t^{(k-1)}) \tag{4}$$

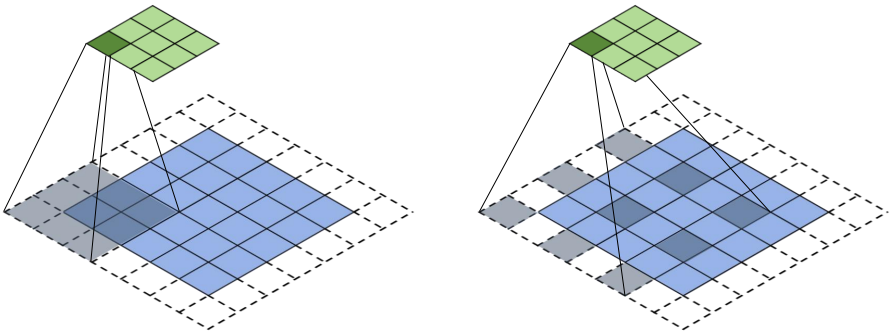

Figure 4. Normal CNN (**left**) acts on a continuous position of the input matrix and continuously slides to do convolution and pooling. The dilated CNN (**right**) adds a dilation width to the filter. When it acts on the input matrix, it skips the input data in the middle of all the dilation width, and the size of the filter matrix itself remains the same, so the filter achieves a wider input.

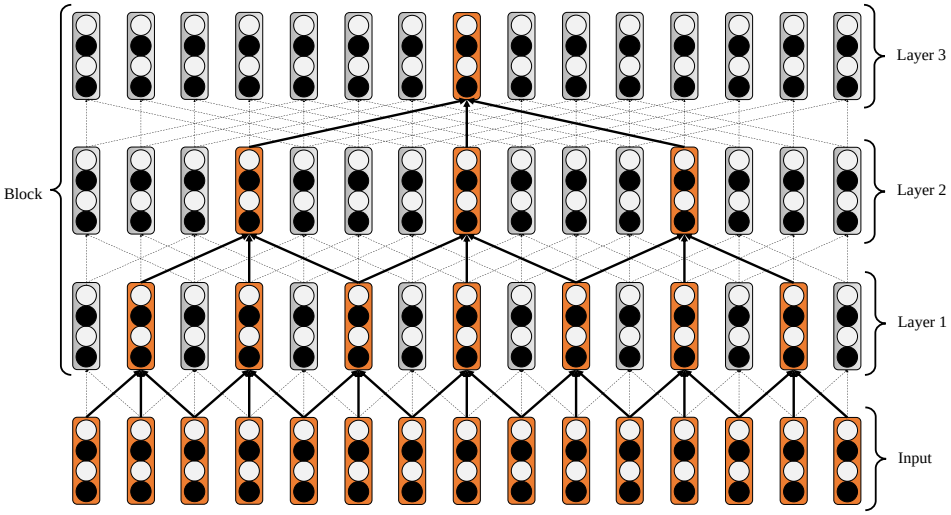

Figure 5. A dilated CNN block. The input is a sequence of texts, each of which is the word representation of Figure 2 (for example, 15 words are entered). The circles just represent the concept of dimensions. Layer 1 has a coefficient of expansion of 1 to obtain a receptive field of 3. Layer 2 expands by 1 on the basis of Layer 1; although the receptive field in Layer 2 is still 3, the receptive field in the input is 7. Take the orange unit of Layer 3 as an example: all related neurons are connected by thick lines, and the receptive field is 15. Each unit of each layer is connected to the three units of the previous layer; the number of parameters remains unchanged, but the receptive field increases geometrically.

3.3.2. Attention Mechanism

Our model can rapidly aggregate broad context by using iterated dilated CNNs. However, the extracted broad context usually ignores the importance of the current word for the current tag.

To alleviate this problem, we applied the attention mechanism to our model. The attention mechanism has recently become popular in image processing and natural language processing. Ling et al. [23] utilized an attention mechanism to enforce tagging consistency across multiple instances of the same token in a document. Bharadwaj et al. [31] introduced the attention mechanism to enhance their model performances. However, their attention mechanism focuses on which encoded elements contribute to the generation of the current unit. Different from them, we applied the attention mechanism to focus on the related tokens in the sequence.

In the attention layer, we calculated the attention weight by Equation (5) between the current token and all tokens in the sequence as the projection matrix and normalized it with the softmax function.

$$\alpha_{t,i} = \frac{exp(sim(b_t^L, b_i^L))}{\sum_{k=1}^{n} exp(sim(b_t^L, b_k^L))} \tag{5}$$

where b_t^L is the final output position t in Equation (4) and sim denotes the similarity between two vectors. The similarity between two vectors was measured by cosine similarity according to Equation (6).

$$sim(b_t^L, b_i^L) = \frac{W_a(b_t^L b_i^L)}{\|b_t^L\| \, \|b_i^L\|} \tag{6}$$

where W_a is a weight matrix, which is learned in the training process. We took the weight vector computation of the unit at position b_t^L as an example to explain the detailed computation process. According to Equation (7), we can achieve the coefficient of each output in the attention layer, then we calculate the output a_t under this attention coefficient.

$$a_t = \sum_{i=1}^{N} \alpha_{t,i} b_i^L \tag{7}$$

Then, the output of the current position and the output of the attention layer are concatenated as the output of this module.

$$o_t = W_o[a_t : b_t^L] \tag{8}$$

where W_o is a weight matrix that maps the output to the category space. Finally, like the general NER method, the CRF layer is added for the final sequence labeling.

3.3.3. Linear Chain CRF

Though the deep learning-based NER method can automatically extract high-level abstract features for each token for category judgment, the rules of transfer among labels tend to be ignored. It has been proven that the correlations among adjacent labels can be very beneficial in sequence tagging [1–3]. The linear chain CRF combines the advantages of the maximum entropy model and the hidden Markov model. It considers the transfer relationships among the labels, enabling the acquisition of the optimal label sequence in the sequence labeling.

We considered E to be the matrix of the score output by our model. The i-th column is the vector o_i obtained by Equation (8). The element E_{i,y_i} of the matrix is the score of the tag y_i of the i-th token in the sentence. We introduced a tagging transition matrix T. The element T_{y_{i-1},y_i} of the matrix is the score of transition from tag y_{i-1} to y_i. This transition matrix will be trained as the parameter of our network.

Therefore, for a sentence $S = \{w_1, w_2, w_3, ..., w_n\}$, the score of its prediction sequence $y = \{y_1, y_2, y_3, ..., y_n\}$ can be expressed as Equation (9):

$$s(S, y) = \sum_{i=1}^{n} E_{i,y_i} + \sum_{i=1}^{n-1} T_{y_{i-1},y_i} \tag{9}$$

Under the condition of the given sentence S, the probability of sequence label y is as follows:

$$p(y|S) = \frac{e^{s(S,y)}}{\sum_{\tilde{y}\in Y_X} e^{s(S,\tilde{y})}} \tag{10}$$

During the training process, the likelihood function of the marker sequence is:

$$log(p(y|S)) = s(S,y) - log(\sum_{\tilde{y}\in Y_X} e^{s(S,\tilde{y})}) \tag{11}$$

where Y_X represents all possible sets of markers, and a valid output sequence can be obtained by the likelihood function. The set of sequences with the highest overall probability is output by Equation (12) when predicting the optimal label:

$$y^* = \arg\max_{\tilde{y}\in Y_X} s(X,\tilde{y}) \tag{12}$$

The loss function was minimized by backpropagation during training, and the Viterbi algorithm was applied to find the tag sequence with maximum probability during testing.

4. Experiments and Analysis

4.1. Datasets

Both CCKS2017 [32] and CCKS2018 [33] (China Conference on Knowledge Graph and Semantic Computing) have published NER tasks for recognizing entities (such as body, symptom, etc.) on CEMR. Table 1 lists the statistics for different categories of entities in the datasets. CCKS2017 Task 2 (Electronic Medical Record Named Entity Recognition) provides a total of 800 real electronic medical record data to identify five types of entities (symptom, check, treatment, disease, body), including 300 labeled data and 500 unlabeled data. Each clinical electronic case was divided into four domains including general items, medical history, diagnosis, and discharge. The CCKS2018 CNER Task 1 provides an annotated corpus of 600 data as the training dataset to identify five types of entities (symptom, description, operation, drug, body).

Table 1. Statistics on different categories of entities in the dataset. CCKS, China Conference on Knowledge Graph and Semantic Computing.

CCKS2017				CCKS2018			
Entity	**Count**	**Train**	**Test**	**Entity**	**Count**	**Train**	**Test**
Symptom	7831	6846	1345	Symptom	3055	2199	856
Check	9546	7887	1659	Description	2066	1529	537
Treatment	1048	853	195	Operation	1116	924	192
Disease	722	515	207	Drug	1005	884	121
Body	10,719	8942	1777	Body	7838	6448	1390
Total	29,866	24,683	5183	Total	15,080	11,984	3096

To train and evaluate the proposed model effectively, we split the dataset into the training set and test set by a ratio of 4:1, In addition, we collected nearly 13,496 medical records from the Internet [28] as training corpora for word vectors.

4.2. Parameter Setting

The parameters of our model are listed in Table 2. We obtained word representation with word embedding dimension 256, char embedding dim 64, POS embedding dim 64, and position embedding dim 128. The filter width, numfilter, dilation, and block number of the iterated dilated CNN module were set to 3, 256, [1,1,2] and 4, respectively. Other parameters can also be seen in Table 2.

To compare with the traditional conditional random field model, we constructed feature templates for the CRF as shown in Table 3. In the feature templates, $U01$ indicates the feature number, and $W[-1,0]$ indicates the word at the previous position of the current word. Similarly, $W[0,0]$ and $W[1,0]$ indicate the words of the current word and the next word.

Table 2. Parameter setting for the modules.

Module	Parameter Name	Value
Word Representation	word embedding dim	256
	char embedding dim	64
	POS embedding dim	64
	position embedding dim	128
Iterated Dilated CNNs	filter width	3
	numfilter	256
	dilation	[1,1,2]
	block number	4
Other	learning rate	1e-3
	dropout	0.5
	gradient clipping	5
	batch size	64
	epoch	40

Table 3. Feature templates of the CRF model.

Word Feature	POS Feature	Description
$U01$:%W[−1,0]	$U06$:%P[−1,0]	previous word (POS)
$U02$:%W[0,0]	$U07$:%P[0,0]	current word (POS)
$U03$:%W[1,0]	$U08$:%P[1,0]	next word (POS)
$U04$:%W[0,0] %W[−1,0]	$U09$:%P[0,0] %P[−1,0]	current word and previous word (POS)
$U05$:%W[0,0] %W[1,0]	$U10$:%P[0,0] %P[1,0]	current word and next word (POS)

4.3. Evaluation Metrics

The accuracy (P), Recall (R), and F1 value proposed at the MUC-6 meeting were used for NER. The specific indicators in the indicator system are as follows: The sample was divided into positive and negative; if the sample was positive, the number predicted by the model as positive was recorded as TP; if the sample was negative, the number predicted by the model as positive was recorded as FP; if the sample was negative and was modeled as suchthe number predicted to be negative was denoted as TN; the number of samples being positive and predicted by the model as negative was denoted as FN.

$$P = \frac{TP}{TP + FP} \tag{13}$$

$$R = \frac{TP}{TP + FN} \tag{14}$$

$$F1 = \frac{2PR}{P + R} \tag{15}$$

Accuracy is the ratio of the number of correctly-identified named entities to the number of named entities identified in the experiment; the recall rate is the ratio of the number of correctly-identified named entities to the total number of named entities in the sample. The F1-value is the harmonic average of the accuracy and recall rate.

4.4. Comparison with Other Methods

To validate the effectiveness of our model fully, we compared our attention-based ID-CNNs-CRF model to previous state-of-the-art methods on the two datasets described in Section 4.1.

As shown in Table 4, our attention-based ID-CNNs-CRF outperformed the prior methods for most metrics. Compared with the ID-CNNs-CRF, our method obtained better performance (improvements of 5.95%, 7.48%, and 7.08% in precision, recall, and F1-score, respectively). This demonstrates that the position embedding and attention mechanism had a huge performance improvement for the ID-CNNs model, since it could fuse word-order information and pay attention to the local context. In addition, our model outperformed the Bi-LSTM-CRF model, showing that our attention-based ID-CNNs-CRF was also an effective token encoder for structured inference. Nevertheless, the attention-based ID-CNNs-CRF model we proposed was slightly higher than the Bi-LSTM-CRF model in the F-score by 0.58% (0.73% in CCKS2018), but the precision was still inferior to the Bi-LSTM-CRF model (CRF in CCKS2018).

Table 4. Results of different NER models on the CCKS2017 and CCKS2018 datasets.

Model	CCKS2017			CCKS2018		
	Precision %	Recall %	F1-score %	Precision %	Recall %	F1-score %
CRF [19]	89.18	81.60	85.11	92.67	72.10	77.58
LSTM-CRF [1]	87.73	87.00	87.24	82.61	81.70	82.08
Bi-LSTM-CRF [2]	94.73	93.29	93.97	90.43	90.49	90.44
ID-CNNs-CRF [9]	88.20	87.15	87.47	81.27	81.42	81.34
Attention-ID-CNNs-CRF(ours)	94.15	94.63	94.55	91.11	91.25	91.17

Our method was not only a better token encoder than the Bi-LSTM-CRF, but it was also faster. Table 5 lists the test time on the test set, compared to the Bi-LSTM-CRF. Our test set contained 512 pieces of data, and each model was tested five times for the average. The average test time for our model and Bi-LSTM-CRF was 12.81 and 15.62, respectively. The model we proposed was 22% faster than Bi-LSTM-CRF. Compared with ID-CNNs-CRF, the speed of our model was slightly worse, but our model outperformed it by a wide margin.

Table 5. Comparison of the test time.

Model (512 Test Data)	Time (s)	Speed
Bi-LSTM-CRF [2]	15.62	1.0×
ID-CNNs-CRF [9]	11.96	1.31×
Attention-ID-CNNs-CRF (ours)	12.81	1.22×

4.5. Comparison of Entity Category

To compare the recognition capabilities in different entity categories comprehensively, we enumerated and compared the F1-score of the three main models (Bi-LSTM-CRF, ID-CNN-CRF without position embedding, and the attention-based ID-CNNs-CRF model we proposed) in five categories. Detailed experimental results are shown in Table 6.

Table 6. Comparison of the entity category on the CCKS2017 and CCKS2018 datasets.

Model	CCKS2017					
	Body	**Check**	**Disease**	**Signs**	**Treatment**	**Average**
Bi-LSTM-CRF [2]	94.81	96.40	87.79	95.82	88.10	93.97
ID-CNNs-CRF [9]	90.89	88.28	79.67	91.04	81.27	87.47
Attention-ID-CNNs-CRF (ours)	95.38	97.79	86.55	96.91	87.64	94.55
Model	**CCKS2018**					
	Body	**Symptom**	**Operation**	**Drug**	**Description**	**Average**
Bi-LSTM-CRF [2]	92.59	93.12	87.43	82.86	86.33	90.44
ID-CNNs-CRF [9]	84.37	85.81	76.59	80.42	84.61	81.34
Attention-ID-CNNs-CRF (ours)	95.18	94.47	85.86	80.06	91.99	91.17

As shown in Table 6, the average F1-score of our model was improved by 0.58% (0.73% in CCKS2018) compared to the Bi-LSTM-CRF while maintaining a 1.22× (Table 5) faster test time speed. The performance of the ID-CNNs-CRF was worse than the other two models in each category. The F-score ranged from 87.79% to 96.40% (from 82.86% to 93.12% in CCKS2018) in different categorized entities when it was computed on our attention-based ID-CNNs-CRF model, whereas the range was from 86.55% to 97.79% (from 80.06% to 95.18%) when it was computed from the Bi-LSTM-CRF model. For most categories, the model we proposed was significantly better than the Bi-LSTM-CRF. For example, it had a slightly high F1-score (2.59% on body and 5.66% on description in CCKS2018) than the Bi-LSTM-CRF model. However, for a few categories, the Bi-LSTM-CRF had a higher F1-score (1.24% on disease and 0.46% on treatment in CCKS2017) than our model. We conjecture that it might be due to the imbalanced data distribution (the number of different entities varied greatly) because the entity number of all these categories was small (as can be seen in Section 4.1).

4.6. Comparison of Different Parameters

On the one hand, the increase of the dimension will increase the complexity of the model, making the model more generalized. On the other hand, the increase of the dimension will bring huge computational cost. To optimize the network structure, we evaluated the impact of four different embedding dimensions on the F1-score. When we evaluated different embeddings, such as word embedding, we fixed the dimensions of other embedding (such as char embedding, etc.) as relatively optimal results. We tried different test embedding dimensions, increasing from 64 to 512 with a step size of 2^n.

From the result shown in Figure 6, the dimension of word embedding had a large impact on the F1-score of the model. The F1-score was 89.56% using 64 dimensions of word embedding and was increased to 93.98% using 256 dimensions of word embedding (CCKS2017). Although the 512-dimensional word embedding was 0.13% higher than the 256-dimensional word embedding F1-score, it doubled the computing resources. The dimensions of char embedding and POS embedding had little effect on the F1-score. In order to have lower computational complexity, we selected 64 as their best dimension number. Position embedding also had an impact on the model performance. We found that the larger the dimension of position embedding, the better the performance of the model. However, the larger the dimension, the weaker it was. To balance model performance and computational cost, we chose a dimension of 256.

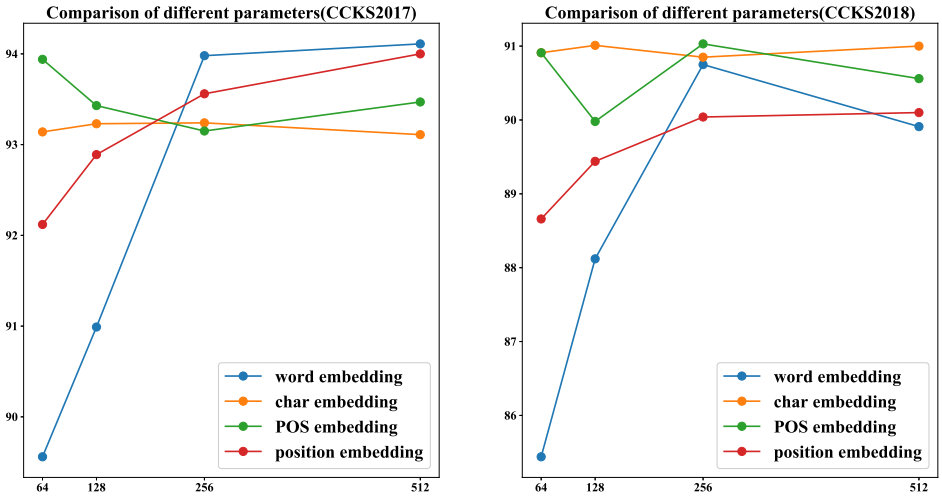

Figure 6. The chart on the left shows the effect of changes in the different embedding dimensions of CCKS2017 on the F1-score of the model (the chart on the right is CCKS2018). The abscissa represents different dimensions, and the ordinate represents the F1-score of the model on the test set.

5. Conclusions

In this paper, we proposed an attention-based ID-CNNs-CRF model for NER on CEMR. Firstly, word representation combined with position embedding was used for the input of our model to capture the word-order information. Secondly, we stacked four dilated CNN blocks to obtain broad semantic information to make up for the shortage of the CNN-based model in the language field. Then, the attention mechanism was applied to pay more attention to the characteristics of the current words and increase the performance of the model. Finally, the CRF was utilized to obtain the optimal tag sequence. The experiments on two CEMR datasets demonstrated that the attention-based ID-CNNs-CRF was superior to state-of-the-art methods with a faster test time. As we know, Bi-LSTM-CRF has a temporal structure, and its output at each step depends on the output of the previous step. This temporal structure is time consuming, especially when dealing with long text. Our model had good parallelism and could make full use of the GPU for parallel computing. Compared with the ID-CNNs-CRF, our model obtained better performance (improvements of 5.95%, 7.48%, and 7.08% in precision, recall, and F1-score, respectively). This demonstrates that the position embedding and attention mechanism had a huge performance improvement for the ID-CNNs model. In addition, our model outperformed Bi-LSTM-CRF, showing that our attention-based ID-CNNs-CRF was also an effective token encoder for structured inference. The model we proposed was 22% faster than the Bi-LSTM-CRF. There was no significant improvement in our model in the number of entities with fewer samples. Therefore, our future work is to study how to improve the recognition of these entities with fewer samples.

Author Contributions: Conceptualization, M.G. and S.W.; Methodology, M.G.; Software, M.G.; Validation, M.G., S.W. and Q.X.; Formal Analysis, K.D.; Investigation, M.G.; Resources, M.G.; Data Curation, M.G.; Writing—Original Draft Preparation, M.G.; Writing—Review & Editing, S.W.; Visualization, S.W.; Supervision, M.G.; Project Administration, M.G.; Funding Acquisition, S.W.

Funding: This research was funded by [National Key Research and Development Program of China] grant number [2017YFC0907505] and [Shanghai Natural Science Foundation] grant number [18ZR1414400].

Acknowledgments: The authors thank all the anonymous reviewers for their insightful comments and useful suggestions.

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

References

1. Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; Dyer, C. Neural architectures for named entity recognition. *arXiv* **2004**, arXiv:1603.01360.
2. Ma, X.; Hovy, E. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv* **2016**, arXiv:1603.01354.
3. Rondeau, M.A.; Su, Y. LSTM-Based NeuroCRFs for Named Entity Recognition. In Proceedings of the Interspeech, San Francisco, SF, USA, 8–12 September 2016; pp. 665–669.
4. Rei, M.; Crichton, G.K.; Pyysalo, S. Attending to characters in neural sequence labeling models. *arXiv* **2016**, arXiv:1611.04361.
5. Yin, W.; Kann, K.; Yu, M.; Schütze, H. Comparative study of CNN and RNN for natural language processing. *arXiv* **2017**, arXiv:1702.01923.
6. Minh, D.L.; Sadeghi-Niaraki, A.; Huy, H.D.; Min, K.; Moon, H. Deep learning approach for short-term stock trends prediction based on two-stream gated recurrent unit network. *IEEE Access* **2018**, *6*, 55392–55404. [CrossRef]
7. Collobert, R.; Weston, J.; Bottou, L.; Karlen, M.; Kavukcuoglu, K.; Kuksa, P. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.* **2011**, *12*, 2493–2537.
8. Wang, C.; Chen, W.; Xu, B. Named entity recognition with gated convolutional neural networks. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*; Springer: Cham, Germany; Basel, Switzerland, 2017; pp. 110–121.
9. Strubell, E.; Verga, P.; Belanger, D.; McCallum, A. Fast and accurate sequence labeling with iterated dilated convolutions. *arXiv* **2017**, arXiv:1702.02098, 138.
10. Hirschman, L.; Morgan, A.A.; Yeh, A.S. Rutabaga by any other name: extracting biological names. *J. Biomed. Inf.* **2002**, *35*, 247–259. [CrossRef]
11. Tsai, R.T.H.; Sung, C.L.; Dai, H.J.; Hung, H.C.; Sung, T.Y.; Hsu, W.L. NERBio: Using selected word conjunctions, term normalization, and global patterns to improve biomedical named entity recognition. In Proceedings of the Fifth International Conference on Bioinformatics, New Delhi, India, 18–20 December 2006.
12. Tsuruoka, Y.; Tsujii, J.I. Boosting precision and recall of dictionary-based protein name recognition. In Proceedings of the ACL 2003 workshop on Natural language processing in biomedicine, Sapporo, Japan, 11 July 2003; pp. 41–48.
13. Yang, Z.; Lin, H.; Li, Y. Exploiting the performance of dictionary-based bio-entity name recognition in biomedical literature. *Comput. Biol. Chem.* **2008**, *32*, 287–291. [CrossRef] [PubMed]
14. Han, X.; Ruonan, R. The method of medical named entity recognition based on semantic model and improved svm-knn algorithm. In Proceedings of the 2011 Seventh International Conference on Semantics, Knowledge and Grids, Beijing, China, 24–26 October 2011; pp. 21–27.
15. Collier, N.; Nobata, C.; Tsujii, J.I. Extracting the names of genes and gene products with a hidden markov model. In Proceedings of the 18th conference on Computational linguistics, Saarbrücken, Germany, 31 July–4 August 2000; pp. 201–207.
16. GuoDong, Z.; Jian, S. Exploring deep knowledge resources in biomedical name recognition. In Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications, Geneva, Switzerland, 28–29 August 2004; pp. 96–99.
17. Chieu, H.L.; Ng, H.T. Named entity recognition with a maximum entropy approach. In Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, Edmonton, AB, Canada, 27 May–1 June 2003; pp. 160–163.
18. Leaman, R.; Islamaj Doğan, R.; Lu, Z. Dnorm: disease name normalization with pairwise learning to rank. *Bioinformatics* **2013**, *29*, 2909–2917. [CrossRef] [PubMed]
19. Kaewphan, S.; Van Landeghem, S.; Ohta, T.; Van de Peer, Y.; Ginter, F.; Pyysalo, S. Cell line name recognition in support of the identification of synthetic lethality in cancer from text. *Bioinformatics* **2015**, *32*, 276–282. [CrossRef] [PubMed]
20. Zhu, Q.; Li, X.; Conesa, A.; Pereira, C. Gram-cnn: A deep learning approach with local context for named entity recognition in biomedical text. *Bioinformatics* **2015**, *34*, 1547–1554. [CrossRef] [PubMed]

21. Xu, K.; Zhou, Z.; Gong, T.; Hao, T.; Liu, W. Sblc: a hybrid model for disease named entity recognition based on semantic bidirectional lstms and conditional random fields. In Proceedings of the 2018 Sino-US Conference on Health Informatics, Guangzhou, China, 28 June–1 July 2018.
22. Chowdhury, S.; Dong, X.; Qian, L.; Li, X.; Guan, Y.; Yang, J.; Yu, Q. A multitask bi-directional rnn model for named entity recognition on chinese electronic medical records. *BMC Bioinform.* **2018**, *19*. [CrossRef] [PubMed]
23. Luo, L.; Yang, Z.; Yang, P.; Zhang, Y.; Wang, L.; Lin, H.; Wang, J. An attention-based bilstm-crf approach to document-level chemical named entity recognition. *Bioinformatics* **2017**, *34*, 1381–1388. [CrossRef] [PubMed]
24. Sang, E.F.; Veenstra, J. Representing text chunks. In Proceedings of the Conference on European Chapter of the Association for Computational Linguistics, Bergen, Norway, 8–12 June 1999.
25. Lai, S.; Liu, K.; He, S.; Zhao, J. How to generate a good word embedding. *IEEE Intell. Syst.* **2016**, *31*, 5–14. [CrossRef]
26. Joulin, A.; Grave, E.; Bojanowski, P.; Mikolov, T. Bag of tricks for efficient text classification. *arXiv* **2016**, arXiv:1607.01759.
27. Le, Q.; Mikolov, T. Distributed representations of sentences and documents. In Proceedings of the International Conference on International Conference on Machine Learning, Beijing, China, 21–26 June 2014.
28. Available online: http://case.medlive.cn/all/case-case/index.html?ver=branch (accessed on 21 August 2019). (In Chinese)
29. Available online: https://github.com/fxsjy/jieba (accessed on 21 August 2019).
30. Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv* **2015**, arXiv:1511.07122.
31. Bharadwaj, A.; Mortensen, D.; Dyer, C.; Carbonell, J. Phonologically aware neural model for named entity recognition in low resource transfer settings. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 1462–1472.
32. Li, J.; Zhou, M.; Qi, G.; Lao, N.; Ruan, T.; Du, J. *Knowledge Graph and Semantic Computing. Language, Knowledge, and Intelligence*; Springer: Singapore, 2018.
33. Zhao, J.; Van Harmelen, F.; Tang, J.; Han, X.; Wang, Q.; Li, X. *Knowledge Graph and Semantic Computing. Knowledge Computing and Language Understanding*; Springer: Singapore, 2019.