International Journal of Computational Intelligence Systems
Vol. **14(1)**, 2021, *pp.* 471–481
DOI: https://doi.org/10.2991/ijcis.d.201216.001; ISSN: 1875-6891; eISSN: 1875-6883
https://www.atlantis-press.com/journals/ijcis/

Research Article

Dual Neural Network Fusion Model for Chinese Named Entity Recognition

Dandan Zhao[1,2], Jingxiang Cao[3], Degen Huang[1,*], Jiana Meng[2], Pan Zhang[2]

[1]*School of Computer Science and Technology, Dalian University of Technology, Dalian, China*
[2]*School of Computer Science and Engineering, Dalian Minzu University, Dalian, China*
[3]*School of Foreign Languages, Dalian Minzu University, Dalian, China*

ARTICLE INFO

Article History
Received 08 July 2020
Accepted 11 Dec 2020

Keywords
Chinese named entity recognition
Dual neural network fusion
Bi-directional long-short-term memory
Self-attention mechanism
Dilated convolutional neural network

ABSTRACT

Chinese named entity recognition (NER) has important effect on natural language processing (NLP) applications. This recognition task is complicated in its strong dependent-relation, missing delimiters in the text and insufficient feature representation in a single model. This paper thus proposes a dual neural network fusion model (DFM) to improve Chinese NER performance. We integrate the traditional bi-directional long-short-term memory (BiLSTM) structure and self-attention mechanism (ATT) with dilated convolutional neural network (DCNN) to better capture context information. Additionally, we exploit the Google's pre-trained model named bi-directional encoder representations from transformers (BERT) as the embedding layer. The proposed model has the following merits: (1) a dual neural network architecture is proposed to enhance the robustness of extracted features. (2) An attention mechanism is fused into the dual neural network to extract implicit context representation information in Chinese NER. (3) Dilated convolutions are used to make a tradeoff between performance and executing speed. Experiments show that our proposed model exceeds the state-of-the-art Chinese NER methods.

1. INTRODUCTION

The named entity recognition (NER) is a foundation task of natural language processing (NLP). NER has very important effect on many fields, such as entity linking (Blanco *et al.* [1]), relation extraction (Lin *et al.* [2]), and question answering (Min *et al.* [3]). The purpose of NER is to determine the boundaries of entities in the text and to classify entity names into different types, such as person names, location names, and institution names. Take sentence "Steve Jobs was a co-founder of Apple Inc.," for example. The task of NER aims to recognize "Steve Jobs," which is a named entity of a person, and "Apple Inc." is a named entity of an organization.

Traditional NER techniques rely on hand-crafted rules. For example, according to the Brill inference rules, a tagger is obtained to deal with speech processing (Kim and Woodland [4]). The similar method is also adopted in the biomedical field. A synonym dictionary from preprocessed phase is used to recognize proteins and genes from a given biomedical text. Due to limitations of specific rules and faulty dictionaries, high-precision and low-recall systems cannot be transplanted to other fields. To improve the generalization of the model, feature engineering was proposed for NER. Specifically, word-level information including morphology and case is recognized via internal and external feature extractors in the NER system (Zhou and Su [5]; Settles [6]; Liao and Veeramachaneni [7]). Other information such as local syntax and global features, text, and corpus features was used to improve the accuracy of the NER task too. Those features are used in a lot of machine learning techniques including hidden markov models (HMMs) (Saito and Nagata [8]), maximum entropy (ME) models (Chieu and Ng [9]), and conditional random fields (CRFs) (Feng *et al.* [10]) to guide the NER application. The earlier NER algorithm IderntiFinder (Bikel *et al.* [11]) analyzed and classified name entities including time expressions, names, dates, and numerical quantities. Later, techniques based on classifiers and optimization methods were proposed to boost the classification results for NER. Zhou and Su [5] used hand-crafted features with conditional probability independence to improve IderntiFinder. Borthwick *et al.* [12] proved that ME was very effective in NER. Carreras *et al.* [13] added a decision tree to binary AdaBoost classifier to boost the expressive capability of NER model. Li *et al.* [14] first proposed a support vector machine (SVM) in CoNLL 2003 to address NER problem. As a variant, Isozaki and Kazawa [15] introduced a faster SVM for NER task. However, these feature-based methods excessively relied on hand-crafted features, which resulted in low efficiency.

Deep learning techniques with powerful expressive abilities become popular for NLP task and achieve good classification performance recently. Different from traditional methods, deep learning is very suitable to automatically learn and find hidden information. Thus, typical methods of deep learning techniques such as convolutional neural network (CNN) have been successfully applied in NER task.

* *Corresponding author. Email: huangdg@dlut.edu.cn*

For example, Huang *et al.* [16] combined a long-short-term memory (LSTM) and CRF into a CNN to mine more accurate features for English NER. To make full use of elements information, Ma and Hovy [17] used an end-to-end BiLSTM-CNNs-CRF architecture to address the NER task. Because traditional CNN has weak ability to extract input features of long sequences, Strubell *et al.* [18] used dilated convolutions to increase the receptive field to alleviate long-distance dependence. However, these methods still lack the ability to learn better representation.

Due to complex sentence meaning and delimiters, Chinese NER task is more difficult. Chinese has no natural delimiters for spaces as in English, which make word boundaries ambiguous. For example, "劳动公园史展馆 (labor park history exhibition hall)" is an organization name entity type in NER task. But the example can be divided into words of "劳动 (labor)," "公园 (park)," "史 (history)," and "展馆 (exhibition hall)"; "劳动公园 (labor park)," "史 (history)" and "展馆 (exhibition hall)"; "劳动公园 (labor park)," "史展馆 (history exhibition hall)"; or "劳动公园史 (labor park history)," "展馆 (exhibition hall)" in different granularity of segmentation as shown in Figure 1. It is difficult to unify the rules of word segmentation. Therefore, it is difficult to identify the named entity of "劳动公园史展馆 labor park history exhibition hall" correctly based on the word recognition model.

Additionally, the word-based model is not valid for unknown words. The flexible rules of Chinese word formation create a large number of out of vocabulary (OOV) words and named entities are a large part of OOV words, so the model based on Chinese characters is more effective.

Another notable point is that Chinese entity names have close connection with the context. For example, "文章" (Zhang Wen) is treated as a named entity of person in the sentence "文章主演雪豹" (Zhang Wen acts the leading role in Snow Leopard). However, "文章" means "article" or "publication" of the nonentity word in Chinese text in general, which makes models hard to learn the context representation.

To resolve those problems, this paper proposes a dual neural network approach for Chinese NER, which merges the traditional bi-directional long-short-term memory (BiLSTM) structure and self-attention mechanism (ATT) with dilated convolutional neural network (DCNN) to capture more context information, such as character expressions from global and local contexts. Additionally, an NLP pretrained model is used to accelerate the convergence speed.

<table>
<tr><td>granularity</td><td colspan="4">劳动公园史展馆
(labor park history exhibition hall)</td></tr>
<tr><td>fine-grained</td><td>劳动
(labor)</td><td>公园
(park)</td><td>史
(history)</td><td>展馆
(exhibition hall)</td></tr>
<tr><td>slightly fine-grained</td><td colspan="2">劳动公园
(labor park)</td><td>史
(history)</td><td>展馆
(exhibition hall)</td></tr>
<tr><td>slightly coarse-grained</td><td colspan="2">劳动公园
(labor park)</td><td colspan="2">史展馆
(history exhibition hall)</td></tr>
<tr><td>coarse-grained</td><td colspan="3">劳动公园史
(labor park history)</td><td>展馆
(exhibition hall)</td></tr>
</table>

Figure 1 Visual table for the influence of word segmentation granularity on named entity recognition (NER).

The proposed network has the following contributions:

1. A dual neural network architecture is proposed to recognize Chinese named entity. The context characteristics of Chinese characters based on bi-directional encoder representations from transformers (BERT) input are further explored through the dual model, which improves the performance of Chinese NER.
2. The training speed of the system is not sacrificed for the improved performance of Chinese NER via combining the DCNN and BiLSTM with ATT.

The rest of this paper is organized as follows: related work is given in Section 2, and the proposed method is illustrated in Section 3. The experiment results and analyses are listed in Section 4. Finally, a conclusion is presented in Section 5.

2. RELATED WORK

2.1. Chinese NER

NER has important applications in the field of NLP. However, Chinese sentences are expressed by string of characters, with no clear delimiters between words. Traditional methods rely on manually defined rules (Kim and Woodland [4]) or hand-crafted features to complete the Chinese NER (Mccallum and Li [19]). Many flexible machine learning methods such as HMMs, SVM, and CRF are used in the field of NER too. However, they need to use complex optimization algorithms to improve the performance of Chinese NER, which was very time-consuming.

The neural network technology was used to handle NER task due to great self-learning capability, the novel techniques are very useful in mining hidden information. According to the characteristics of the task, deep network can treat NER as sequence labeling task, which consists of three parts: distributed representations for input, context encoder, and tag decoder (Li *et al.* [20]).

The input representation is classified into word-based and character-based. Collobert and Weston [21] designed a word-based NER model to extract orthographic and lexicons and dictionaries information. Zhai *et al.* [22] exploited segmentation and labeling to conduct a CNN model for sequence chunking where SENNA is fused into the CNN to improve the accurate rate of the NER task. Because Chinese word segmentation is compulsory for those models, the models above all suffer from segmentation errors for Chinese NER. In character-based models (Ma and Hovy [17]) applied a deep network to obtain character-level expression, which used a preprocessor to obtain character expression vector. The vector concatenated with the given word embedding is used as input of context encoder in a recurrent neural network (RNN). He and Sun [23] took the position of characters into account. Peters *et al.* [24] proposed a deep bi-directional language model (biLM), where character convolutions and the top two-layers in the biLM were computed as word representation. Although great improvements have been achieved by these methods on NER task, the existing model are not satisfactory in obtaining either useful boundary information from Chinese word segment (CWS) or type

information for NER. The character-level feature representation can be considered as the input of the model, which will effectively avoid the error accumulation and noise caused by the inaccurate word segmentation to the NER task.

Combining context encoder and CNN is an effective scheme for NER. Collobert *et al.* [25] applied a tag from the whole sentence to conduct a deep network for NER application. Additionally, changing the network architecture is popular in NER. Strubell *et al.* [18] gathered multiple DCNN as well as iterated dilated convolutional neural network (IDCNN) to capture more context information, which was useful to predict NER results. To better model sequential information, Lample *et al.* [26] used BiLSTM to encode sequence context information, which applied transformer-based and self-attention method to further improve accuracy and efficiency.

As the final phase of NER model, tag decoder can use context-related expression as input and obtain a sequence of label corresponding to the input label. Since NER task strongly depends on output tags, many models also use a CRF layer to act as the tag decoder (i.e., Zheng *et al.* [27]).

2.2. Attention Methods

Due to its ability to capturing distinctive feature, attention mechanism has achieved excellent results on multiple tasks, such as NLP and image processing (Seo *et al.* [28], Tian *et al.* [29]). Specifically, attention mechanism fused into deep network can improve the ability to deal with a subset of the input information. Thus, using attention mechanism can obtain most representative elements for NER.

Attention mechanisms of different types have been widely used to address NER problem. Rei *et al.* [30] integrated attention method and character-based representation to dynamically determine the categories of information, which was superior to popular NER methods. Additionally, combining prior knowledge and attention idea is useful. Zhang *et al.* [31] fused pictures from Tweets as prior knowledge into adaptive attention mechanism for recognizing text task, but Tweets' resources are legally limited. Zukov-Gregoric *et al.* [32] used self-attention method to deal with a single sequence rather than related two sequences, which can improve the training speed. Xu *et al.* [33] used an attention technique to urge more document-level features for NER. It is seen that attention methods are very suitable to NER task.

2.3. DCNN-Related Models

Extracting low-level and suitable features has important influences on NLP application, for example, sentence classification (Kim [34]), emotion analysis (Dos Santos and Gatti [35]), speech processing (Abdel-Hamid *et al.* [36]). To make a tradeoff between performance and speed, traditional CNNs rely on pooling function to reduce the feature map to improve the execution time. That may cause information loss. To address this issue, Lei *et al.* [37] fused dilated convolutions into CNN to enlarge the receptive field via neighboring words to capture information that is more useful. After that, more and more CNN depending on dilated convolutions of different dilated factors to obtain better skills in NLP were proposed. For example, Oord *et al.* [38] used dilated convolutions in speech recognition and machine translation.

From previous researches, we found that dilated convolutions had been widely utilized in NLP task. Thus, we use DCNN for Chinese NER in this paper.

3. METHOD

Assume an input sentence is $X = \{x_1, x_2, \cdots x_N\}$, where $x_i \in R^d$ is the i-th character vector representation in the input sentence X, d is the character vector dimension and N represents the length of this sentence. Similarly, the sentence ground-truth is defined as $Y = \{y_1, y_2, \cdots y_N\}$, where y_i represents the i-th character's label in the set of all possible label solutions. The purpose of this paper is to learn a mapping $f_\theta : X \rightarrow Y$ that uses a NER model to obtain the entity tags of all characters in the input text. To better capture the representation of features in sentences, a dual neural network is proposed for Chinese NER. The network architecture of the proposed method is shown in Figure 2. The first layer is an embedding layer using pre-trained BERT, which contains 12 layers of the encoder part in the transformer architecture. The middle encoding layer is composed of two deep neural networks, including the BiLSTM+ATT model and the DCNN model. BiLSTM+ATT utilizes the output of the embedding layer as input. BiLSTM captures the long-range dependency between the characters in sentences, and ATT mechanism is implemented on the output of BiLSTM to better capture the degree of dependency between characters. DCNN, which contains three layers, also utilizes the output of the embedding layer as input and gets the character's context from another process. Then, we use some fusion methods to fuse the output of BiLSTM+ATT with the output of DCNN to get the representation containing the combined features. Finally, the fused representation will be input to the CRF layer to get the corresponding tags. Each layer from designed network is illustrated in the following sections.

3.1. BERT Embedding Layer

The first layer is an embedding layer, which can effectively transform a sequence of characters into dense vectors. To exploit the prior knowledge obtained through pre-training, we use BERT as the embedding layer. The pretrained BERT model uses a bi-directional transformer as the encoder, so that the representation of each character can merge its information with that of its left and right characters. Overall pretraining and fine-tuning procedures for BERT is given in Figure 3. The BERT model does not utilize typical left-to-right or right-to-left language models to deal with pretrained operations. Alternatively, the BERT model is pretrained by utilizing two unsupervised tasks. One is masked language model (MLM), which can be used to train a deep bi-directional representation. 15% of the input tokens are randomly masked, and then the masked tokens are predicted when training. The other task is next sentence prediction (NSP), which aims to learn a model for understanding relations among different sentences. To be specific, the "isNext" relationship of two input sentences are predicted as "yes" or "no," denoting if the second sentence is the next sentence of the first sentence or not, which enables the model to understand the relations of long sequence contexts. After pretraining stage, semantic information is integrated into the model. In the proposed model, the pre-trained BERT model is utilized as embedding layer to get a sequence of dense vectors, and to improve training efficiency. We freeze the BERT layer when training. Given input representation is expressed

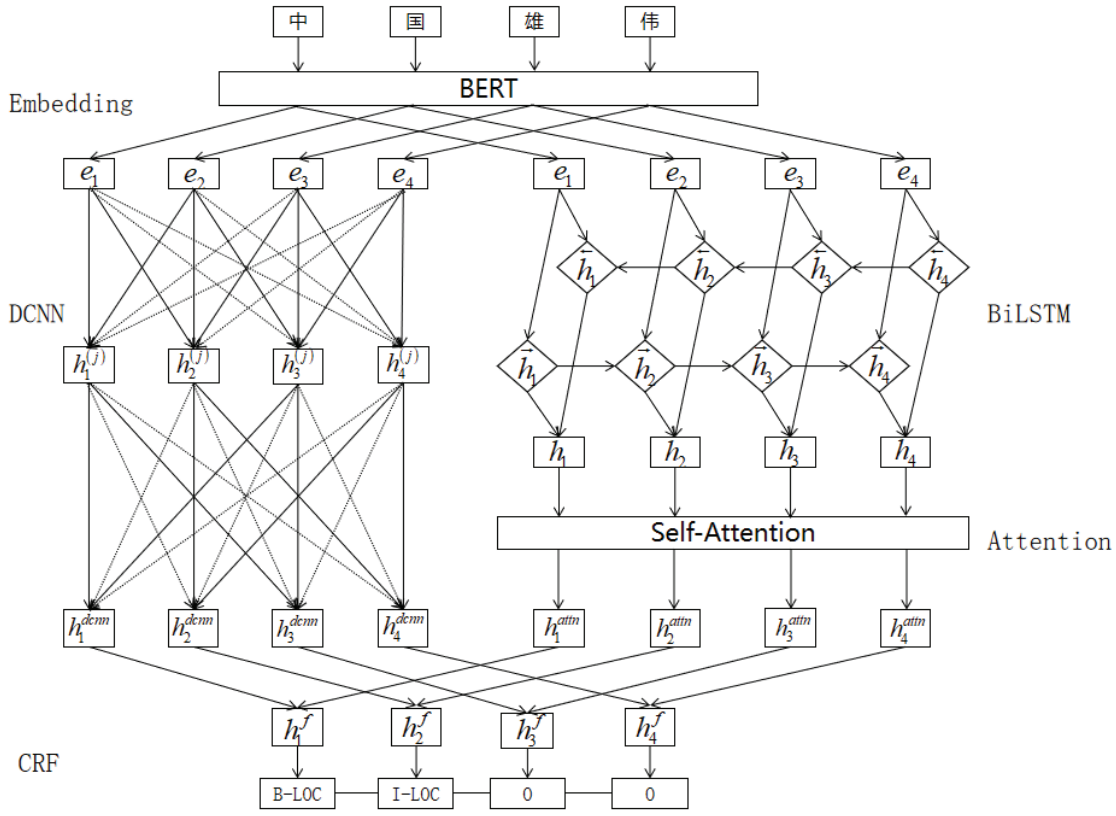

Figure 2 The architecture of dual neural network fusion model (DFM). The embedding layer uses bi-directional encoder representations from transformers (BERT), whose output is the input to the dual encoding layer, BiLSTM+ATT and dilated convolutional neural network (DCNN) layer. The decoder layer is conditional random field (CRF), whose input is the concatenate of the dual layer's output.

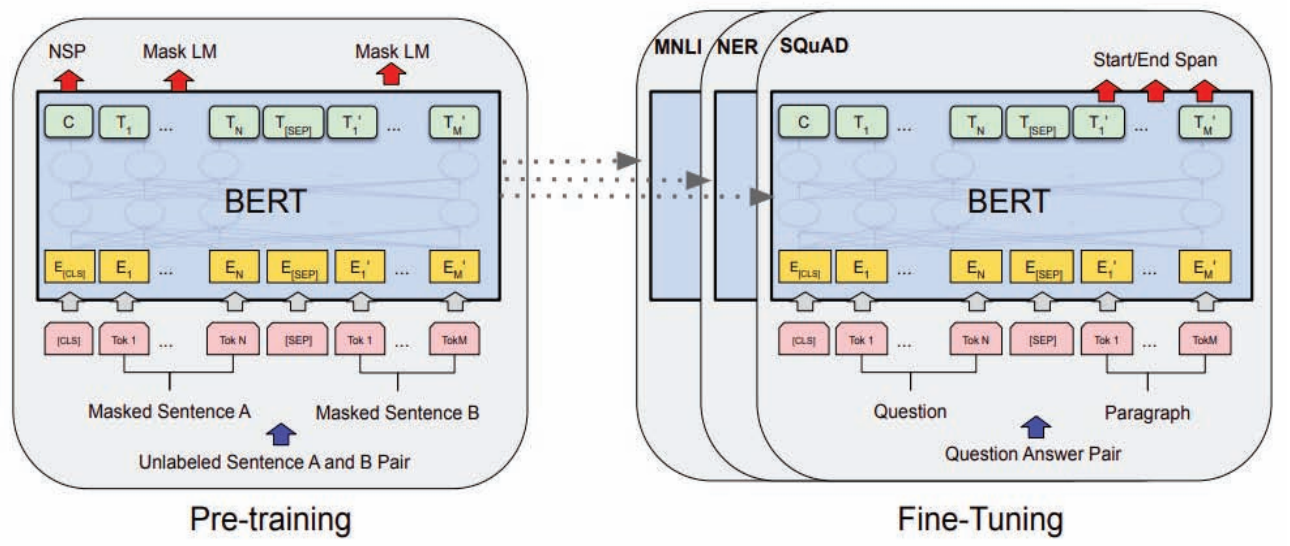

Figure 3 Overall pre-training and fine-tuning procedures for bi-directional encoder representations from transformer (BERT).

as x_i for each character, and the output of the BERT embedding layer, denoted by e_i is computed as follows:

$$e_i = BERT(x_i) \tag{1}$$

Compared with typical language algorithms, the BERT pretrained language model can fully use the features on the left and right sides of Chinese character to achieve a better-distributed expression of the character.

3.2. Dual Model Fusion Encoding Layer

To better capture the feature representation of the sentence, we use dual model fusion method to encode the embedding. One is BiLSTM with ATT, which can better capture the long-range feature information, and the other is a DCNN, which can better capture the local feature information in the context sequence. Then, the outputs of the two networks are fused to obtain the vector representation of Chinese characters. By this method, we capture a better characterization of Chinese text.

3.2.1. BiLSTM model

LSTM is typical operation of popular RNN, which can capture long-range sequence features and deal with sequential data. The LSTM cell is mainly composed of three cell gates and a cell memory which outputs two states, the cell state and the hidden state, to the next cell. LSTM can capture long-range dependencies through three gating mechanisms, that is, input gate, forget gate, and output gate. The input gate determines the percentage of the saved to cell state from the current state. The forgetting gate determines which information in the previous step needs to be retained. The output gate is used to output percentage of information from the internal state of the memory unit at the current moment to determine the value of the next hidden state. The structure of a LSTM cell is shown in Figure 4.

The LSTM cell is computed as follows:

$$\begin{aligned} i_t &= \sigma\left(W_i \cdot [h_{t-1}, e_t] + b_i\right) \\ f_t &= \sigma\left(W_f \cdot [h_{t-1}, e_t] + b_f\right) \\ o_t &= \sigma\left(W_o \cdot [h_{t-1}, e_t] + b_o\right) \\ \tilde{C}_t &= \tanh\left(W_c \cdot [h_{t-1}, e_t] + b_c\right) \\ C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t \\ h_t &= o_t * \tanh\left(C_t\right) \end{aligned} \tag{2}$$

where σ is sigmoid function for activation, $*$ is the dot multiplication, i_t, f_t, and o_t denote input gate, forget gate, and output gate, e_t is the input vector, W_i, W_f, W_o, and W_c denote the connection matrixes, b_i, b_f, b_o, and b_c are bias vectors, $\tilde{C}_t$ denotes the candidate output value, C_t denotes memory unit, and h_t denotes hidden output vector.

Firstly, both left and right contexts are exploited to recognize named entities. Then, BiLSTM is applied to mine hidden expression of characters from global context by the following way:

$$\left[\overrightarrow{h_1}, \overrightarrow{h_2}, ..., \overrightarrow{h_N}\right] = \overrightarrow{LSTM}\left([e_1, e_2, ..., e_N]\right) \tag{3}$$

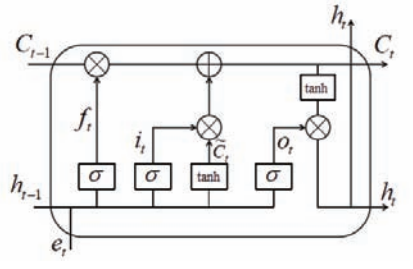

Figure 4 | Structure of a long-short-term memory (LSTM) cell.

$$\left[\overleftarrow{h_1}, \overleftarrow{h_2}..., \overleftarrow{h_N}\right] = \overleftarrow{LSTM}\left([e_1, e_2, ..., e_N]\right) \tag{4}$$

where e_i denotes the i-th character embedding after BERT layer and the h_i denotes the output of forward and backward LSTM. The hidden expression of the i-th character is concatenated by h_i.

$$h_i = \left[\overrightarrow{h_i}, \overleftarrow{h_i}\right] \tag{5}$$

Finally, the output of BiLSTM layer is defined as $h = [h_1, h_2, \cdots, h_N]$, where $h_i \in R^{2S}$ and S denotes the dimension of hidden states in LSTM.

3.2.2. Self-attention mechanism

A self-attention method is merged into BiLSTM network architecture. Attention is a complex cognitive function indispensable to human beings, which refers to the ability of people to ignore some information while paying attention to some information. In Figure 5, the self-attention method is used to establish a relationship between a query and key-value pairs. The query, values, keys, and output are saved as vectors. Specifically, the output can be obtained by the weighted total of the values, where the weight is computed via the given compatibility operation on the query and key. We exploit self-attention to learn the degree of dependency between any two characters in a sentence and to capture information about the inner structure of the sentence. In this paper, we adopt self-attention.

The output of the BiLSTM layer is $h = [h_1, h_2, \cdots, h_N]$, we first perform a linear transformation to get query, key and value computed as follows:

$$\begin{aligned} q_i &= W_q \cdot h_i + b_q \\ k_i &= W_k \cdot h_i + b_k \\ v_i &= W_v \cdot h_i + b_v \end{aligned} \tag{6}$$

where q_i, k_i, v_i denote the query, key, and value vectors, and W_q, W_k, W_v, b_q, b_k, b_v are learning parameters. To better handle the sentence-level information, the output of the self-attention layer is a weighted total of the values, after generating the query, key, and value,

$$h_i^{attn} = \sum_{j=1}^{N} \alpha_{i,j} v_j \tag{7}$$

where $i = 1, 2, \cdots, N$, N denotes the total number of all the characters in the input sentence, v_j denotes the j-th value vector, and α_{ij} is calculated as follows:

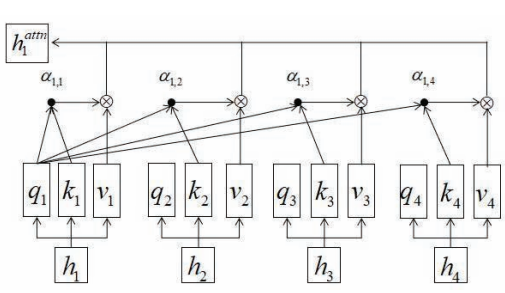

Figure 5 | An example of self-attention mechanism.

$$\alpha_{i,j} = \frac{exp\left(s\left(q_i, k_j\right)\right)}{\sum_{l=1}^{N} exp\left(s\left(q_i, k_l\right)\right)} \tag{8}$$

The score function s is defined as follows:

$$s\left(q_i, k_j\right) = v^T \tanh\left(W_q q_i + W_k k_j\right) \tag{9}$$

where v, W_q, W_k are learning parameters.

3.2.3. DCNN model

CNN has wide applications in image processing and NLP (Tian *et al.* [29]). The role of the convolution operation is to highlight features and to extract more features that are obvious, which improves the efficiency of model both in training and in testing. Inspired by DCNN (Strubell *et al.* [18]), we propose using context-module of stacked DCNN to further increase receptive field to capture more context. Figure 6 is used to denote a dilated CNN module with dilation width of 4 and filter width of 3. It shows that with the increase of depth, the width of the dilation convolution will increase without loss of resolution or increase of parameters. Thus, stacked dilated CNN can capture global features in a whole sentence and document. Inspired by that, we use DCNN to obtain better representations of Chinese characters. In this paper, DCNN is set after BERT embedding layer, its input sequence is denoted as $e = [e_1, e_2, \cdots, e_N]$. The j-th dilated convolutional layers with dilation factor h^{δ} are expressed as $D(j)^{\delta}$. Dilation-1 convolution is set in the first layer, which transforms the embedding representation to a hidden representation $h^{(1)}$:

$$h^{(1)} = D_1^{(0)}(e) \tag{10}$$

Then, the output of L layers of dilated convolutions are utilized as input of the former layer with ReLU activation function:

$$h^{(j+1)} = \mathrm{ReLU}\left(D_{j+1}^{(j)}\left(h^{(j)}\right)\right) \tag{11}$$

where the $j \in \{1, 2, \cdots, L-1\}$ and the final output is denoted as follows:

$$h^{dcnn} = h^{(L)} \tag{12}$$

3.3. CRF Decoding Layer

Considering the dependencies between continuous labels, we use the CRF layer for sequential tagging. It denoted the self-attention layer output as $h^{attn} = \left[h_1^{attn}, h_2^{attn}, \cdots, h_N^{attn}\right]$ and the DCNN layer output as $h^{dcnn} = \left[h_1^{dcnn}, h_2^{dcnn}, \cdots, h_N^{dcnn}\right]$. The input of the CRF layer is the hidden representations of characters obtained by the dual models and it is calculated by

$$h^f = Concat\left(h^{attn}, h^{dcnn}\right) \tag{13}$$

Given the label sequence $Y = \{y_1, y_2, \cdots, y_N\}$, the conditional probability of label sequence Y with input h fusion is obtained,

$$P\left(Y|h^f; \theta\right) = \frac{\prod_{i=1}^{N} \phi\left(h_i^f, y_i, y_{i-1}\right)}{\sum_{y' \in Y(s)} \prod_{i=1}^{N} \phi\left(h_i^f, y_i', y_{i-1}'\right)} \tag{14}$$

where $Y(s)$ is the set of all possible label sequences of sentence s, and $\varphi\left(h_i^f, y_i, y_{i-1}\right)$ is the score function, which is computed as follows:

$$\phi\left(h_i^f, y_i, y_{i-1}\right) = \exp\left(y_i^T W h_i^f + y_{i-1}^T T_{yi}\right) \tag{15}$$

where W and T denote parameters in the CRF layer. In training a model, a negative log-likelihood function is used as objective function. Given training examples $\{X_i, Y_i\}_{i=1}^{K}$, the objective function L is expressed as

$$L = -\sum_{i=1}^{k} \log P\left(Y_i|X_i\right) \tag{16}$$

4. EXPERIMENTS

4.1. Experimental Datasets

Two public datasets are used to test performance of our model in different domains. In news domain, we use Microsoft Research Asia (MSRA) NER dataset of SIGHAN Bakeoff 2006 (Levow [39]). For more varieties in test domains, Chinese Resume dataset (Zhang and Yang [40]) is used to design experiments. The Chinese resume dataset contains eight types of named entities: CONT (Country), EDU (Educational Institution), LOC, PER, ORG, PRO (Profession), RACE (Ethnicity Background), and TITLE (Job Title). The

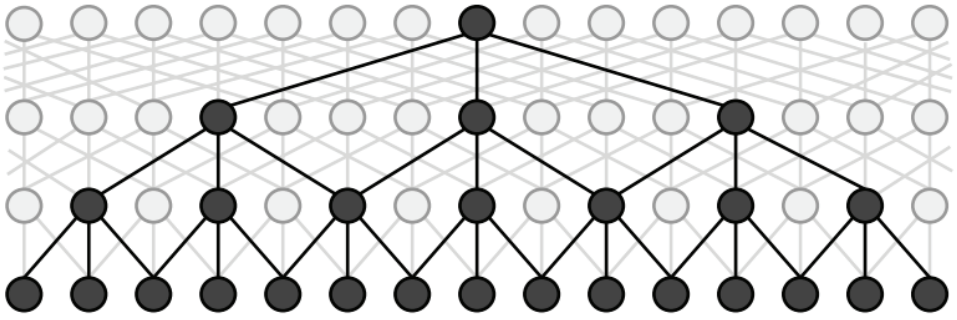

Figure 6 | A dilated convolutional neural network (DCNN) block with maximum dilation width of 4 and filter width of 3.

MSRA NER dataset is composed of ORG, PER, and LOC. We use BIO annotation mode in experiments. The details of the two datasets are shown in Table 1.

4.2. Experimental Settings

We utilize the tokenizer and character embedding pre-trained by BERT (Devlin *et al.* [41]) released by Google. To speed up training and verify our methods, the parameters of BERT are frozen to reduce the impact of the pretrained language model when training the model. The experimental setting is shown in Table 2.

4.3. Evaluation Indicator

To evaluate the performance of Chinese NER, we use the Precision (P), Recall(R), and *F1*-score as metrics. The computing formulas are 17, 18, and 19.

$$P = \frac{TP}{TP + FP} \times 100\% \tag{17}$$

$$R = \frac{TP}{TP + FN} \times 100\% \tag{18}$$

$$F = \frac{2PR}{P + R} \times 100\% \tag{19}$$

where *TP* is the number of entities correctly identified by the model, *FP* is the number of unrelated entities identified by the model, and *FN* is the number of related entities but not detected by the model.

4.4. Parameter Tuning

We use **BERT-Base,** Chinese Simplified and Traditional version, 12-layer, 768-hidden, 12-heads. The dimensionality of BiLSTM hidden states is 128. For self-attention strategy, the initial number of heads is 6.

We adjust hyper-parameter according to the result of F1 value for Chinese NER task. The results of hyper-parameter adjustment experiment on dropout on MSRA dataset are shown in Figure 7. The network structure generated randomly is the best when dropout is 0.5, and F1 value is the highest at the same time, so we choose 0.5 as the dropout value in the experiment.

The results of hyper-parameter adjustment experiment on batch size are shown in Figure 8. With the same amount of data, increasing the batch size can reduce the number of iterations needed to run an epoch and further accelerate the processing speed. However, with the increase of batch size, more and more epoch will be used to achieve the same precision, and too large batch size will easily lead to poor generalization performance and slow convergence (no convergence within 200 epochs). In our experiment, the performance is better when the batch size is set to multiples of 16 than that of 10. In addition, the number of epoches set in our experiment is small, and the results show that the effect is better when the batch size is set to 32.

Additionally, the dimensionality of the BiLSTM hidden states is 128, the kernel sizes of $K = 3$ and $L = 3$ are set in the DCNN with dilation rates of $\delta = 1, 1, 2$. To prevent over fitting problem, we apply dropout operation with a rate of 0.5. The parameters are shown in Tables 3 and 4.

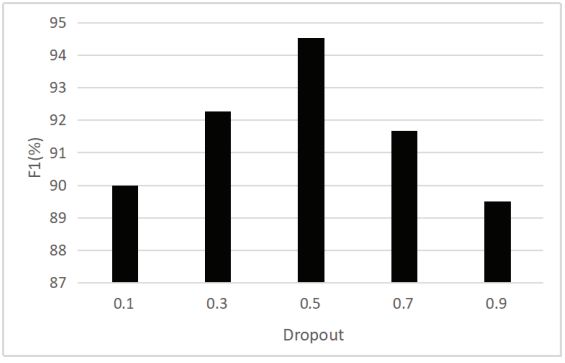

Figure 7 The impact of dropout on the experimental results.

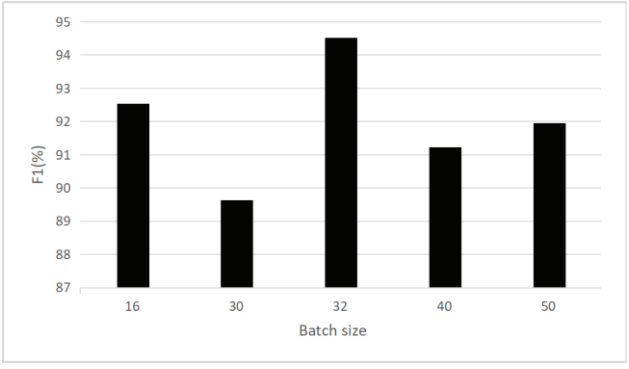

Figure 8 The impact of batch size on the experimental results.

Table 1 The statistics of datasets.

Datasets	Train Sent	Dev Sent	Test Sent
Chinese resume	3821	463	477
MSRA NER dataset	46364	–	4365

MSRA, Microsoft Research Asia.

Table 2 The experimental setting.

Operating System	Ubuntu
GPU	GeForce RTX 2080 Ti
CPU	Intel (R) Xeon (R) CPU E5-2640 v4 @2.40GHz
CUDA	10.2.89
CUDNN	7.6.5
Python	3.6.10
Tensorflow	1.14.0

Table 3 Parameters for dilated convolutional neural network (DCNN).

	Filters	kernal_size	dilation_rate
First layer	64	3*3	1
Second layer	128	3*3	1
Third layer	128	3*3	2

4.5. Experimental Results and Analyses

In this part, the experimental results of the proposed deep networks will be shown and analyzed on the Chinese Resume dataset and MSRA dataset, respectively.

We assume that our proposed model as dual neural network fusion model (**DFM**), *-LSTM* as DFM without BiLSTM+ATT, and *-DCNN* as DFM without DCNN. As shown in Table 5, our proposed method has better performance than other existing methods on the Chinese Resume dataset. We compare with three state-of-the-art models, Lattice model (Zhang and Yang [40]), CAN Model (Zhu and Wang [42]), and WC-LSTM + longest (Liu *et al.* [43]), and our training model is superior to reference methods above, where it can achieve F1-score of 96.41% without referring to extra information. It proves that the proposed method is very effective for Chinese NER.

When evaluating on MSRA, we use several existing methods as comparative methods to conduct experiments for Chinese NER as shown in Table 6. Chen *et al.* [44], Zhang *et al.* [45], and Zhou *et al.* [46] leverage rich hand-crafted features and Dong *et al.* [47] introduce radical features into LSTM-CRF. Yang *et al.* [48] gave the combination of CNN, Bi_LSTM, and CRF-based character to extract stroke embedding and *n*-gram features for Chinese NER. The model of WC-LSTM (Liu *et al.* [43]) and CAN-NER (Zhu and Wang [42]) in 2019 are also carried out on MSRA dataset. These show that proposed techniques are very useful for Chinese NER. Additionally, we achieved F1-score of 94.52%.

Table 4 Hyper-parameters for long-short-term memory (LSTM).

Hyper-parameters	Number
rnn_units	128
dropout	0.5
batchize	32
Learning rate	0.01

Table 5 Results on Chinese Resume dataset.

Models	P	R	F1
Lattice [40]	94.81	94.11	94.46
CAN Model [42]	95.05	94.82	94.94
WC-LSTM + longest [43]	95.27	95.15	95.21
DFM(ours)	**96.57**	**96.37**	**96.41**
–LSTM	94.02	96.66	95.09
–DCNN	94.19	94.34	94.27

Table 6 Results on MSRA NER dataset.

Models	P	R	F1
Chen *et al.* [44]	91.22	81.71	86.20
Zhang *et al.* [45]	92.20	90.18	91.18
Zhou *et al.* [46]	91.86	88.75	90.28
Dong *et al.* [47]	91.28	90.62	90.95
Yang *et al.* [48]	92.04	91.31	91.67
Zhang and Yang [40]	93.57	92.79	93.18
Liu *et al.* [43]	94.36	92.38	93.36
Zhu and Wang [42]	93.53	92.42	92.97
DFM(ours)	**94.58**	**94.47**	**94.52**
–LSTM	94.49	93.29	93.88
–DCNN	90.63	91.48	90.78

To verify the effectiveness of each module in the model, ablation experiments are carried out on MSRA dataset and the results are shown in Figure 9 and Table 7.

We use word vector as input instead of BERT output in -BERT model. After the introduction of Bert, the F1 value in DFM increases by 14.07% than DCNN_BiLSTM_ATT_CRF model, which indicates that the BERT has better semantic information expression.

Compared with the BERT-DCNN-BiLSTM-CRF model, the F1 value of the DFM model increases by 1.84% after the introduction of ATT. The F1 value of the DCNN-BiLSTM-ATT-CRF model increases by 3.13% compared with that of the DCNN-BiLSTM-CRF model after the introduction of ATT. It shows that ATT has a good performance in extracting important semantic information with the better understanding of global semantic information.

BERT-DCNN-CRF model performs 3.1% better than BERT-BiLSTM-ATT-CRF, which shows that DCNN is effective. After the introduction of BiLSTM and ATT, the F1 value of DFM increases by 0.64%, indicating that the context semantic relationship can be obtained more accurately by combining the local feature information obtained by DCNN model and the global feature information obtained by BiLSTM and ATT.

The experimental results show that the BERT, DCNN, BiLSTM, and ATT all play an important role in the model.

4.6. Experimental Efficiency

In this part, we mainly illustrate the efficiency of the proposed method. Firstly, we compare of the number parameters the model used to show the speed of our model. To further prove the efficiency of the proposed model, we design comparative experiments in terms of training time and convergence speed. Due to the space limitation, the test experiments are designed only on MSR dataset. We still use **DFM** to represent our dual neural network fusion

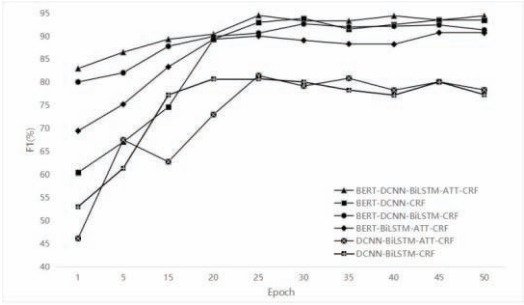

Figure 9 Ablation experiment results.

Table 7 The best results of ablation experiments.

Ablation Part	Model Component	F1
DFM(whole)	BERT-DCNN-BiLSTM-ATT-CRF	94.52
–LSTM	BERT-DCNN-CRF	93.88
–ATT	BERT-DCNN-BiLSTM-CRF	92.68
–DCNN	BERT-BiLSTM-ATT-CRF	90.78
–BERT	DCNN-BiLSTM-ATT-CRF	80.45
–BERT&ATT	DCNN-BiLSTM-CRF	77.32

method. *-LSTM* indicates DFM without BiLSTM+ATT and *-DCNN* represents DFM without DCNN. The parameters of the model are given in Table 8.

Although the number of total parameters in our model is relatively large, the parameters that our model needs to train are very small. The total number of parameters reaches 102M, but 101M of the parameters do not require training, which means that we do not need to train too many parameters to achieve stat-of-the-art results. This can greatly reduce the training time and improve the training efficiency of the trained model.

The training time of each model from different epochs is listed in Table 9. The batch size of 32 is conducted on MSRA dataset. The details of MSRA dataset are given in Table 2 and the run-time of per epoch and per step from models are shown in Table 9.

The training time per step of our model is roughly the sum of the two models. It achieves fast convergence and high efficiency.

5. CONCLUSION AND FUTURE WORK

In this paper, we design a dual neural network fusion method to improve the performance of the Chinese NER model. Firstly, the character-level feature representation of the input data is obtained through the pretraining model BERT, and then a more accurate feature representation is used to represent the context sequence to guide the design of the Chinese NER network. Secondly, the dual network extracts features more powerfully for Chinese NER. It can better capture the remote feature information and local feature information in the context sequence. Thirdly, integrating the ATT into the proposed network can extract features that are more implicit and obtain the dependencies between arbitrary characters, so that the model can better handle character-level feature representations and obtain faster training speed. In addition, dilated convolution is added to the proposed network to capture more contextual information, which can also reduce computational cost. For the Chinese NER task, our method has better performance on data sets in different fields than the references.

Our future work is as follows: Firstly, we will improve the model by adding Chinese stroke features to make it effective for traditional Chinese character recognition. Secondly, the current research is limited to the entities in sentences. In the future, experiments will be conducted to study how the captured character-level feature representation can better represent word-level and sentence-level feature representation, and then consider extending the recognition range to between sentences and within the text range. Finally, the model will be applied to the recognition of OOV and the NER task in Chinese classical literature.

Table 8 The parameters of the models.

Model	Total Parameters	Trainable Parameters	Nontrainable Parameters
DFM(ours)	102,075,890	693,746	101,382,144
–LSTM	101,598,002	215,858	101,382,144
–DCNN	101,580,222	198,078	101,382,144

Table 9 Time per epoch and per step of models.

Model	Time (s)/Epoch	Time (ms)/Step
DFM(ours)	801	22
–LSTM	318	9
–DCNN	331	9

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

AUTHORS' CONTRIBUTIONS

All authors contributed to the work. All authors read and approved the final manuscript.

ACKNOWLEDGMENTS

Authors sincerely thank the anonymous reviewers for their valuable suggestions that have greatly enhance the quality of the work. The work is supported by National Key Research and Development Program of China No. 2020AAA0108004; Scientific Research Fund of Liaoning Provincial Education Department No .LJYT201906; the National Natural Science Foundation of China under grant No. 61672127, 61772250; and the National Youth Science Foundation of China No. 62006108.

REFERENCES

[1] R. Blanco, G. Ottaviano, E. Meij, Fast and space-efficient entity linking for queries, in Proceeding of 8th ACM International Conference on Web Search and Data Mining, Shanghai, China, 2015, pp. 179–188.
[2] Y.K. Lin, S.Q. Shen, Z.Y. Liu, H.B. Luan, M.S. Sun, Neural relation extraction with selective attention over instances, in Proceeding of 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 2016, pp. 2124–2133.
[3] S. Min, V. Zhong, R. Socher, C. Xiong, Efficient and robust question answering from minimal context over documents, in Proceeding of 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 1725–1735.
[4] J.H. Kim, P.C. Woodland, A rule-based named entity recognition system for speech input, in Proceeding of 6th International Conference on Spoken Language Processing, Beijing, China, 2000, vol. 1, pp. 528–531.
[5] G. Zhou, J. Su, Named entity recognition using an HMM-based chunk tagger, in Proceeding of 40th Annual Meeting on Association for Computational Linguistics, Philadelphia, PA, USA, 2002, pp. 473–480.
[6] B. Settles, Biomedical named entity recognition using conditional random fields and rich feature sets, in Proceeding of International Joint Workshop on Natural Language Processing in Biomedicine and its Applications, Geneva, Switzerland, 2004, pp. 107–110.
[7] W.H. Liao, S. Veeramachaneni, A simple semi-supervised algorithm for named entity recognition, in Proceeding of NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing, Stroudsburg, PA, USA, 2009, pp. 58–65.
[8] K. Saito, M. Nagata, Multi-language named-entity recognition system based on HMM, in Proceeding of ACL 2003 Workshop

on Multilingual and Mixed-language Named Entity Recognition, Sapporo, Japan, 2003, pp. 41–48.
[9] H.L. Chieu, H.T. Ng, Named entity recognition with a maximum entropy approach, in Proceeding of Seventh Conference on Natural Language Learning at HLT-NAACL 2003, Edmonton, Canada, 2003, pp. 160–163.
[10] Y.Y. Feng, L. Sun, Y.H. Lv, Chinese word segmentation and named entity recognition based on conditional random fields models, in Proceeding of 50th SIGHAN Workshop on Chinese Language Processing, Sydney, Australia, 2006, pp. 181–184.
[11] D.M. Bikel, R. Schwartz, R.M. Weischedel, An algorithmthat learns what's in a name, Mach. Learn. 34 (1999), 211–231.
[12] A. Borthwick, J. Sterling, E. Agichtein, R. Grishman, Nyu: description of the MENE named entity system as used in MUC-7, in Proceeding of 7th Message Understanding Conference, Fairfax, VA, USA, 1998.
[13] X. Carreras, L. Marquez, L. Padr, Named entity extraction using AdaBoost, in Proceeding of 6th Conference on Natural Language Learning 2002, Taipei, Taiwan, 2002, pp. 1–4.
[14] Y. Li, K. Bontcheva, H. Cunningham, SVM based learning system for information extraction, in: J. Winkler, M. Niranjan, N. Lawrence (Eds.), Deterministic and Statistical Methods in Machine Learning, Springer, Heidelberg, Berlin, Germany, 2004, pp. 319–339.
[15] H. Isozaki, H. Kazawa, Efficient support vector classifiers for named entity recognition, in Proceeding of 19th International Conference on Computational Linguistics, Stroudsburg, PA, USA, 2002, vol. 1, pp. 1–7.
[16] Z.H. Huang, W. Xu, K. Yu, Bidirectional LSTM-CRF models for sequence tagging, arXiv: Computation and Language, 2015.
[17] X.Z. Ma, E. Hovy, End-to-end sequence labeling via bi-directional LSTM-CNNS-CRF, in Proceeding of 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 2016, pp. 1064–1074.
[18] E. Strubell, P. Verga, D. Belanger, A. Mccallum, Fast and accurate entity recognition with iterated dilated convolutions, in Proceeding of 2017 Conference on Empirical Methods in Natural Language Processing, Comenhagen, Denmark, 2017, pp. 2670–2680.
[19] A. Mccallum, W. Li, Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons, in Proceeding of seventh Conference on Natural language learning at HLT, Stroudsburg, PA, USA, 2003, pp. 188–191.
[20] J. Li, A.X. Sun, J.L. Han, C.L. Li, A survey on deep learning for named entity recognition, IEEE Trans. Knowl. Data Eng. 99 (2020), 1.
[21] R. Collobert, J. Weston, A unified architecture for natural language processing: deep neural networks with multitask learning, in Proceeding of 25th International Conference on Machine Learning, Helsinki, Finland, 2008, pp. 160–167.
[22] F.F. Zhai, S. Potdar, B. Xiang, B.W. Zhou, Neural models for sequence chunking, in Proceeding of Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 2017, pp. 3365–3371.
[23] H.F. He, X. Sun, A unified model for cross-domain and semi-supervised named entity recognition in Chinese social media, in Proceeding of 31nd AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 2017, pp. 3216–3222.
[24] M.E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep contextualized word representations, in Proceeding of 2018 Conference of the North American Chapter of the Association for Computational Linguistics, New Orleans, LA, USA, 2018, pp. 2227–2237.
[25] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, P.P. Kuksa, Natural language processing (almost) from scratch, J. Mach. Learn. Res. 12 (2011), 2493–2537.
[26] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, C. Dyer, Neural architectures for named entity recognition, in Proceeding of 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, 2016, pp. 260–270.
[27] S.C. Zheng, F. Wang, H.Y. Bao, Y.X. Hao, P. Zhou, B. Xu, Joint extraction of entities and relations based on a novel tagging scheme, in Proceeding of 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 1227–1236.
[28] M. Seo, A. Kembhavi, A. Farhadi, H. Hajishirzi, Bidirectional attention flow for machine comprehension, in Proceeding of International Conference on Learning Representations, Toulon, France, 2017.
[29] C.W. Tian, L.K. Fei, W.X. Zheng, Y. Xu, W.M. Zuo, C.W. Lin, Deep learning on image denoising: an overview, Neural Netw. 131 (2020), 251–275.
[30] M. Rei, G.K. Crichton, S. Pyysalo, Attending to characters in neural sequence labeling models, in Proceeding of 26th International Conference on Computational Linguistics, Osaka, Japan, 2016, pp. 309–318.
[31] Q. Zhang, J.L. Fu, X.Y. Liu, X.J. Huang, Adaptive co-attention network for named entity recognition in tweets, in Proceeding of 32nd AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2018, pp. 5674–5681.
[32] A. Zukov-Gregoric, Y. Bachrach, P. Minkovsky, S. Coope, B. Maksak, Neural named entity recognition using a self-attention mechanism, in Proceeding of 29th IEEE International Conference on Tools with Artificial Intelligence, Boston, MA, USA, 2017, pp. 652–656.
[33] G.H. Xu, C.Y. Wang, X.F. He, Improving clinical named entity recognition with global neural attention, in Proceeding of Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data, Macau, China, 2018, pp. 264–279.
[34] Y. Kim, Convolutional neural networks for sentence classification, in Proceeding of 2014 Conference on Empirical Methods in Natural Language Processing, Doha, Qatar, 2014, pp. 1746–1751.
[35] C.D. Santos, M. Gatti, Deep convolutional neural networks for sentiment analysis of short texts, in Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, Dublin, Ireland, 2014, pp. 69–78.
[36] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, D. Yu, Convolutional neural networks for speech recognition, IEEE/ACM Trans. Audio Speech Lang. Process. 22 (2014), 1533–1545.
[37] T. Lei, R. Barzilay, T. Jaakkola, Molding CNNs for text: non-linear, non-consecutive convolutions, Diana Univ. Math. J. 58 (2015), 1151–1186.
[38] A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, K. Kavukcuoglu, WaveNet: a generative model for raw audio, arXiv: 1609.03499v2, 2016.

[39] G.A. Levow, The third international Chinese language processing bakeoff: word segmentation and named entity recognition, in Proceeding of 50th SIGHAN Workshop on Chinese Language Processing, Sydney, Australia, 2006, pp. 108–117.

[40] Y. Zhang, J. Yang, Chinese NER using lattice LSTM, in Proceeding of 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 1554–1564.

[41] J. Devlin, M.W. Chang, K. Lee, K. Toutanova, Bert: pre-training of deep bidirectional transformers for language understanding, in Proceeding of 2019 Conference of the North American Chapter of the Association for Computational Linguistics, Minneapolis, MN, USA, 2019, pp. 4171–4186.

[42] Y.Y. Zhu, G.X. Wang, Can-ner: convolutional attention network for chinese named entity recognition, in Proceeding of 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, MN, USA, 2019, pp. 3384–3393.

[43] W. Liu, T.G. Xu, Q.H. Xu, J.Y. Song, Y.R. Zu, An encoding strategy based word-character LSTM for Chinese NER, in Proceeding of 2019 Conference of the North American Chapter of the Association for Computational Linguistics, Minneapolis, MN, USA, 2019, pp. 2379–2389.

[44] A. Chen, F.C. Peng, R. Shan, G. Sun, Chinese named entity recognition with conditional probabilistic models, in Proceeding of 50th SIGHAN Workshop on Chinese Language Processing, Sydney, Australia, 2006, pp. 173–176.

[45] S.X. Zhang, Y. Qin, W.J. Hou, X.J. Wang, Word segmentation and named entity recognition for SIGHAN bakeoff3, in Proceeding of 50th SIGHAN Workshop on Chinese Language Processing, Sydney, Australia, 2006, pp. 158–161.

[46] J.S. Zhou, W.G. Qu, F. Zhang, Chinese named entity recognition via joint identification and categorization, Chinese J. Electron. 22 (2013), 225–230.

[47] C.H. Dong, J.J. Zhang, C.Q. Zong, M. Hattori, H. Di, Character-based LSTM-CRF with radical-level features for Chinese named entity recognition, in Proceedings of 5th CCF Conference on Natural Language Processing and Chinese Computing, NLPCC 2016, and 24th International Conference on Computer Processing of Oriental Languages (ICCPOL 2016), Kunming, China, 2016, pp. 239–250.

[48] F. Yang, J.H. Zhang, G.S. Liu, J. Zhou, C. Zhou, H.R. Sun, Five-stroke based CNN-BiRNN-CRF network for Chinese named entity recognition, in Proceedings of 7th CCF International Conference on Natural Language Processing and Chinese Computing, Hohhot, China, 2018, pp. 184–195.