Token Relation Aware Chinese Named Entity Recognition

ZEYU HUANG, State Key Laboratory of Software Development Environment, Beihang University, China
WENGE RONG[*], School of Computer Science and Engineering, Beihang University, China
XIAOFENG ZHANG, Sino-French Engineer School, Beihang University, China
YUANXIN OUYANG, School of Computer Science and Engineering, Beihang University, China
CHENGHUA LIN, Department of Computer Science, University of Sheffield, United Kingdom
ZHANG XIONG, School of Information Technology Management, University of International Business and Economics, China

Due to the lack of natural delimiters, most Chinese Named Entity Recognition (NER) approaches are character-based and utilize an external lexicon to leverage the word-level information. Although they have achieved promising results, the latent words they introduced are still non-contextualized. In this paper, we investigate three relations, i.e, adjacent relation between characters, character co-occurrence relation between latent words, and dependency relation among tokens, to address this issue. Specifically, we first establish the local context for latent words and then propose a masked self-attention mechanism to incorporate such local contextual information. Besides, since introducing external knowledge such as lexicon and dependency relation inevitably brings in some noises, we propose a gated information controller to handle this problem. Extensive experimental results show that the proposed approach surpasses most similar methods on public datasets and demonstrates its promising potential.

CCS Concepts: • **Information systems → Web mining**.

Additional Key Words and Phrases: Chinese NER, dependency relation, character co-occurrence, character adjacency, gated mechanism

1 INTRODUCTION

Named Entity Recognition (NER) is the task to identify designators belonging to predefined semantic types such as person and location [26]. It is a fundamental and indispensable task for a wide range of downstream natural language processing (NLP) applications including entity linking [2, 36, 53], question answering [22, 37], information retrieval [1] and relation extraction [4, 30], etc.

Compared to English NER, Chinese NER is more complicated due to the lack of explicit delimiters between words [61]. Thus a lot of character-based approaches have been developed [33, 55]. But it still needs to be supplemented with word-level information. To this end, many existing character-based methods first obtain

[*] Corresponding Author

Authors' addresses: Zeyu Huang, zy2024104@buaa.edu.cn, State Key Laboratory of Software Development Environment, Beihang University, Beijing, China; Wenge Rong, w.rong@buaa.edu.cn, School of Computer Science and Engineering, Beihang University, Beijing, 100191, China; Xiaofeng Zhang, xiaofeng_z@buaa.edu.cn, Sino-French Engineer School, Beihang University, Beijing, China; Yuanxin Ouyang, oyyx@buaa.edu.cn, School of Computer Science and Engineering, Beihang University, Beijing, China; Chenghua Lin, c.lin@sheffield.ac.uk, Department of Computer Science, University of Sheffield, Sheffield, United Kingdom; Zhang Xiong, xiongz@uibe.edu.cn, School of Information Technology Management, University of International Business and Economics, Beijing, China.

2375-4699/2022/4-ART $15.00
https://doi.org/https://doi.org/10.1145/3531534

the lattice structure by matching the latent words from an external lexicon and then leverage their information. The methods for leveraging can be various, for example, Ma et al. [34], Zhang et al. [58] tried to calculate a weighted sum of latent word embedding, Liu et al. [32], Zhang and Yang [60] developed additional dynamic model structures. Recently, Li et al. [28] proposed the FLAT model, which implemented a specific position encoding to represent the lattice structure while utilizing Transformer as encoder and achieved the new state-of-the-art performance.

But there is still some room to improve. First, same as most previous works, the latent word embedding is still non-local contextualized (the same word in different contexts shares the same representation), whereas the comprehension of Chinese words heavily depends on its local context [14]. For example, the meaning of the word “苹果” varies when the context changes from “吃苹果 (eat apple)” to “苹果手机 (iPhone)”. Besides, the main dynamic word embedding such as BERT [9] are token-level (character for Chinese) and are not suitable for encoding latent words, which induces the contextualization of latent word embedding a challenging task [31]. Second, the Transformer’s fully-connected attention is directly utilized for encoding, thus the encoder treats all characters and latent words equally and probably attends to some irrelevant words. This requires the model to learn to select useful relations from many redundant ones.

Therefore, in this research, our goal is to answer the following two questions: (1) how to establish the local context for latent words, and (2) how to help the model select the useful relations among tokens. To these ends, we investigated three relations, i.e, adjacent relations between characters, character co-occurrence relations between latent words, and dependency relations among tokens. Among them, the latter two are helpful when constructing the local context for latent words and all three of them are devoted to strengthening the connections between those strongly related tokens. Fig. 1 is an example, the white rectangles represent the characters, the yellow ones denote the latent words, green arrows mean adjacent relation, red arrows mean character co-occurrence relation and black arrows mean dependency relations. For the latent word “南京市”, its local context is composed by “大桥” (dependency relation), “南京” and “市长” (character co-occurrence relation).

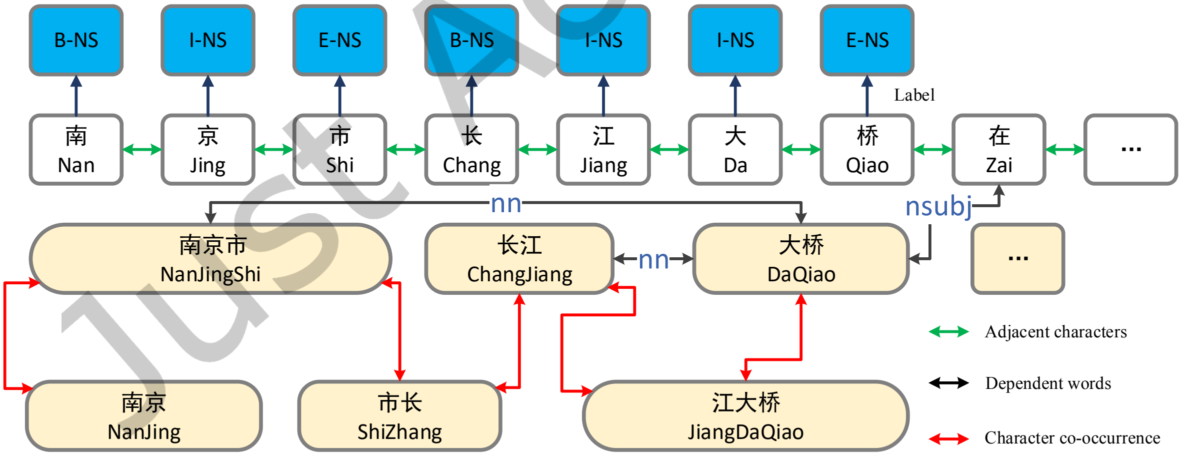


Fig. 1. Examples of Various Relations Among Characters and Words in Chinese

The reasons why we focus on these three relations are as follows. 1) The adjacency relation among characters is critical for Chinese NER since the meanings of a Chinese character heavily depend on its adjacent characters [42]. For example, the character “市” has totally different sense in the word “市长 (Mayor)” and the word “市场

(Market)". Thus a lot of efforts have been devoted to understanding such correlation [45, 62]. 2) The character-occurrence relation is also worth further investigation because different latent words that contain the same characters may be semantically relevant. For instance, latent words "市长 (Mayor)" and "南京市 (NanJing City)" contain the same character "市", which represents similar sense in both words. Several previous approaches make use of the latent words' information ("市长", "南京市") to supplement the embedding of the correspondent character ("市") [11, 18, 34, 46], whereas they ignored that the latent words could also complement each other. 3) The dependency relation is important as well since it provides crucial hints for detecting entity boundaries [8, 43]. Taking Fig.1 for an example, the word "长江 (Yangtze River)" and the word "大桥 (Bridge)" are linked by the "nn" (noun compound modifier) relation, illustrating that the word "长江 (Yangtze River)" modifies the word "大桥 (Bridge)" and "长江大桥 (Yangtze River Bridge)" should be annotated as one named entity.

However, integrating these relations across different levels into the character-based Chinese NER model is challenging. Taking the dependency relation as an example, the previous works that utilize dependency results to enhance their NER model are almost word-level [20, 38]. But the word-based model either requires gold word segmentation results as input or needs to do word segmentation task first, which could cause the error propagation [61]. To address this issue, we adopt FLAT's approach for representing the lattice structure and propose a masked attention mechanism to leverage the relationships across different levels. Specifically, the attention score is solely calculated between the tokens connected with the three aforementioned relations. The proposed mechanism could help the model more effectively learn by masking some immaterial tokens and underlining the important ones. Besides, introducing these relations inevitably brings in some harmful noises. As such we exploit a gated information controller and fuzzification modeling to reduce the impact of noise. First, most Chinese NER datasets lack gold results for dependency parsing and using an external lexicon probably introduces irrelevant words as well. Though it is easy to implement by regarding the extra feature as gold, such approaches lack the ability to filter deleterious information [29, 34, 44, 48, 58], hence we propose a gated information controller for counter-weighting the harmful effect caused by misleading relations. Second, most existing Chinese NER methods treat different relations separately [11, 38, 46]. This may augment the impacts of wrong information and cut off the intrinsic link between different relations. Thus, we conduct a fuzzification when modeling different relations. Due to the inevitable inaccuracy for dependency parsing, we do not distinguish the directions and the types of dependency relations, because sometimes the type or the direction is inaccurately parsed whereas the relation does exist. We also do not distinguish these three relations either as we want to decrease the impact of misleading relations by increasing the related tokens.

The main contributions of our work can be summarized are as follows: 1) We have investigated three specific relations to establish the local context for latent words and to help model select the useful relations among tokens. 2) To the best of our knowledge, we are the first to incorporate the dependency relations into a character-based Chinese NER model. 3) We propose a mask attention mechanism to leverage three relations and exploit two strategies to cope with the noise problem caused by incorrect relations. 4) The extensive experimental results show that our model outperforms many state-of-the-art methods. The ablation studies demonstrat the effectiveness of each proposed component[1].

The rest of the paper is organized as follows: Section 2 presents the basic concept and analyzes previous approaches for Chinese NER. Section 3 elaborates our proposed framework in detail. Section 4 illustrates experimental studies. Section 5 concludes the paper and point out several potential future directions.

2 RELATED WORK

Named entity recognition (NER) is a fundamental task for NLP. It is usually treated as a sequence tagging or a classification task of the input tokens [60]. Earlier NER systems apply the Machine Learning algorithms such as

[1] Our source code is available at https://github.com/gregory-huang/TRAMA/

Hidden Markov Model [13] and CRF [23], but these systems largely depend on hand-crafted features. With the recent development of neural networks, researchers turned to the end-to-end deep neural network to obtain more abstract and convenient representations for NER. Some researchers used Convolution Neural Network (CNN) and obtained comparable results to statistic methods [5, 12]. Later on, to better encode the sequential properties of words and to learn information from contexts [41], the Bidirectional Long Short-Term Memory network-CRF (BiLSTM-CRF) [19] has been developed and has shown promising results. Numerous researchers employed it as the backbone of their models [15, 24]. However, the BiLSTM layer still suffers from seizing global contextual information [27]. Besides, due to its sequential nature, it is inconvenient for parallelization. It is thus inefficient.

To tackle these two problems, Transformer-based frameworks [47, 52] have been proposed to replace the LSTM-based architectures owing to their capability of parallelization and their advantage in modeling long-range contexts. The Transformer is widespread in various NLP tasks such as pre-trained models [9] or language modeling [35], while its performance has still room to improve in the NER task [17]. For better utilization of Transformer [10, 56], many efforts have been devoted to incorporating the appropriate position embedding into the attention mechanism. Instead of using the entire Transformer model, certain researchers combine the LSTM and the self-attention for taking into account both sequential features and long-range context information [21, 57].

The models mentioned above, e.g., CNN, Transformer, BiLSTM, etc, are referred to as the context encoder in the NER system. From this perspective, there is not much difference between Chinese NER and general English NER. What differs Chinese NER from the other European language NER task is the lack of natural word delimiter (such as blank space and capitalization in English), whereas the Chinese named entity boundaries frequently correspond to word boundaries. Therefore, it is intuitive to apply Chinese Word Segmentation first and then treat it as a sequence labeling task. This kind of method suffers from two limitations, i.e., the word segmentation errors will propagate to the sequence labeling step and the word or the named entities could be Out of Vocabulary (OOV). Therefore, a lot of character-based approaches have been developed and are dominated in the Chinese NER problem [33, 55]. In order to compensate for the loss of word-level information in the character-based model, the introduction of external knowledge into the model has become a crucial step for the Chinese NER system. The external knowledge mainly includes gazetteers [11, 49], syntactic constituents [50], Part-of-Speech tag [3], dependency relations [20] and lexicon. Among them, the lexicon has been attached much attention.

To cope with the features brought by latent words matched with the lexicon, a portion of previous works aim at obtaining adaptive embedding. The major idea is to calculate a weighted sum of latent word embedding and to fuse it with correspondent character embedding. The method of calculation and fusion varies considerably. For obtaining the weight of each latent word, the attention mechanism is a popular choice [58, 59]. Contrary to the dynamic weight of the attention mechanism, some other researchers utilized the statistic weight depending on word frequency [34].

Another trend is to design a dynamic architecture that is compatible with the reception of the latent words as additional input. In the beginning, the lattice-LSTM [60] was proposed to use an extra word cell for integrating latent word embedding. However, since the lattice structure in each sentence is quite different, it is less possible to optimize the model in batches. Target on this problem, LR-CNN [15] was proposed to capture multi-gram information by stacking the multiple layers of CNN. It eventually accelerated 3.21 times faster than lattice-LSTM. Later on, due to the natural graphic characteristic of lattice-structure, Graph Neural Network (GNN) based frameworks, such as CGN [46], NMM [11], LGN [16], were proposed. They first used LSTM to capture sequential information for character and then employed GNN to model lattice structure. Recently, FLAT [28] was developed, which proposed a type of position encoding to represent the lattice structure. However, similar to the earlier approaches, the latent word embedding used in FLAT is still non-contextualized. And the noise problem brought by introducing irrelevant latent words in the lexicon was not been taken care of. Following FLAT, we adopted its position encoding to represent the lattice structure and employed masked attention and gate mechanism as encoder to extend the model's performance.

Furthermore, the dependency relation among various tokens has also drawn much attention for solving general NER tasks. Several methods [20, 38] have been proposed to employ dependency relations since such relations provide key information to recognize named entities. But introducing dependency relations into character-based Chinese NER model has rarely been studied and is still challenging as the dependency parsing result is among words. This deficiency motivates us to propose an effective approach to incorporate the dependency relations into the character-based Chinese NER.

3 METHODOLOGY

3.1 Framework Overview

The overall architecture of the proposed token relation aware Chinese NER framework is illustrated in Fig. 2. The proposed framework is composed of a context encoder and a label decoder. Our context encoder consists of three layers, i.e., a token relation aware masked self-attention layer, a gated information controller and a fully connected self-attention layer.

First, we use a lexicon to obtain the flat-lattice structure [28]. Each element in flat-lattice can be defined as a span that includes a token (either character or word) and a tuple indicating the head and the tail of the token. For latent words, The head and the tail denote the position index of the word's the first and the last character in the original sequence. For characters, the head and the tail are both its own position index. Second, we propose the Token Relation Aware Masked Self-Attention layer to incorporate the three relations mentioned. As shown in Fig. 2, the concentric circles represent the neural network unit (gray background for characters and white for words), the solid circles represent the gated mechanism, the arrows express the direction of information flows. The self-attention layer employed composes with two kinds of units. The Query-Key-Value Projection unit is to project the input into three different feature spaces known as Query, Key and Value. Afterwards the attention unit calculates a weighted sum of Values according to the matching score between Keys and Queries. In our situation, the neurons only attend to their adjacent neurons (green arrows), the inter-dependent neurons (red arrows) and neurons sharing the same character (purple arrows). Third, a gated information controller is used to adjust the ratio of the information learned in second step and the raw features obtained from embedding layers. Afterwards we utilize a fully connected multi-head self-attention layer to capture the global information. Finally, a standard conditional random field (CRF) layer [23] is applied to decode the labels.

Formally, the input of model consists of two sub-series, character sequence $C = \{c_1, \dots, c_j, \dots, c_n\}$ and latent words $W = \{w_1, \dots, w_k, \dots, w_m\}$. It should be noted that the directly connecting sequence W cannot reconstruct the original sequence C because one character may appear in more than one latent words. Therefore, rich lexical information is introduced here by considering all of the possible latent words for character c_i.

3.2 Token Encoding Layer

The input of the proposed framework contains two kinds of tokens, i.e., characters and words, here we denote the input sequence of tokens as

$$X_i = \{x^c_{i,1}, x^c_{i,2}, \dots, x^c_{i,n}, x^w_{i,1}, x^w_{i,2}, \dots, x^w_{i,m}\} \tag{1}$$

where $x^c_{i,j} \in \mathbb{R}^{d_e}$ represents the character embedding, $x^w_{i,k} \in \mathbb{R}^{d_e}$ represents the word embedding, d_e is the embedding dimension. The characters embedding are obtained by first using BERT [40] and then being fed into a feed forward neural network $\{W_c, b_c\}$ to transform its dimension to d_e. Similarly, the words embedding are obtained by first using pre-trained word embedding [60] and then being transformed into dimension d_e by

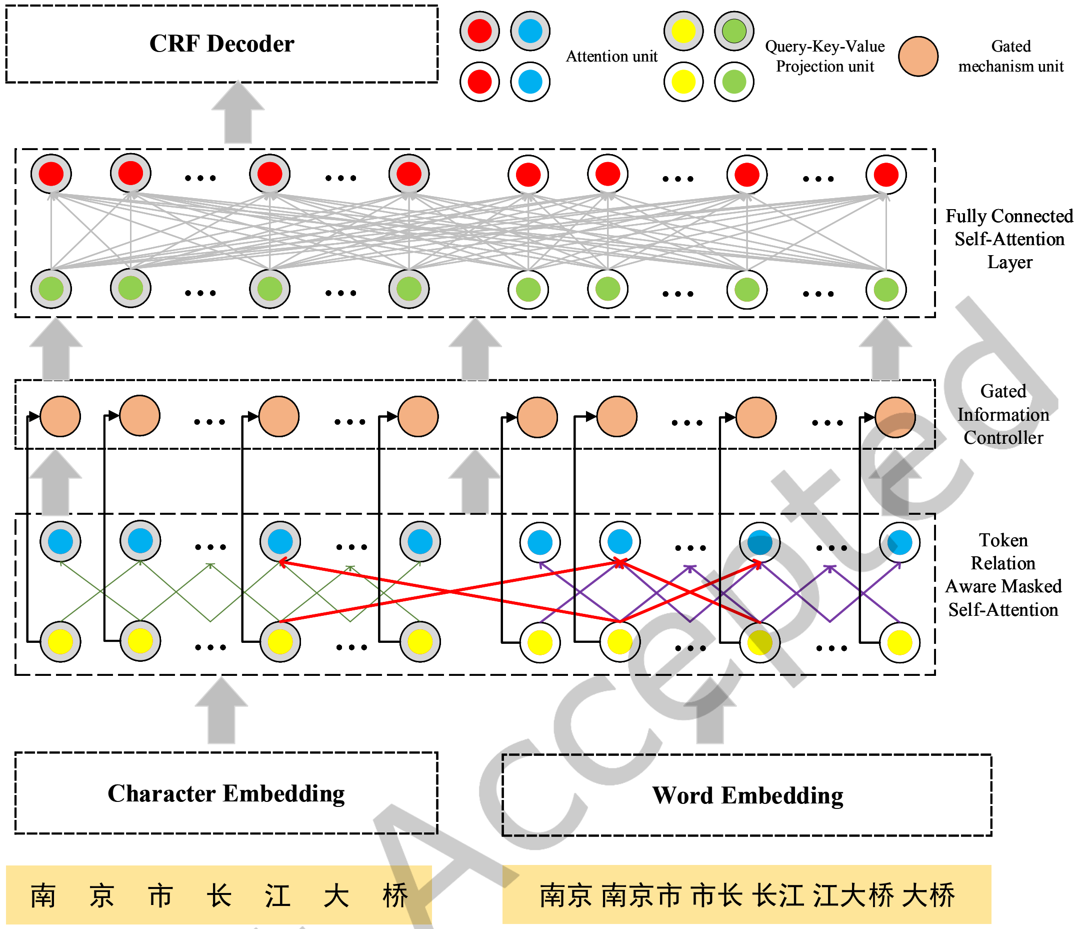

CRF Decoder
Attention unit
Query-Key-Value Projection unit
Gated mechanism unit
Fully Connected Self-Attention Layer
Gated Information Controller
Token Relation Aware Masked Self-Attention
Character Embedding
Word Embedding
南 京 市 长 江 大 桥
南京 南京市 市长 长江 江大桥 大桥

Fig. 2. The Overall Framework Architecture

another feed-forward neural network $\{W_w, b_w\}$:

$$x^c_{i,j} = W_c \cdot \mathbf{BERT}(c_j) + b_c \tag{2}$$

$$x^w_{i,k} = W_w \cdot \mathbf{e}(w_k) + b_w \tag{3}$$

where W_c, W_w and b_c, b_w are trainable parameters, $\mathbf{e}$ is a pre-trained word embedding lookup table. We use token sequence $S = \{s_1, s_2, \ldots, s_{n+m}\}$ to avoid differentiating characters and words.

3.3 Token Relation Aware Masked Self-Attention

For the sake of integrating diverse relations among tokens, a token relation aware masked self-attention layer is developed in our framework.

To better illustrate the relation structure, we define a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the ensemble of graph nodes and $\mathcal{E}$ is the set of graph edges. Each token $s_i \in S$ becomes a graph node. If the tokens s_i and s_j are connected by one of the three kinds of relation aforementioned, we construct an undirected edge $e_{i,j} \in \mathcal{E}$ between those two tokens. Then we can define the set $\mathcal{N}_i$ that contains all neighbor nodes of token s_i.We adopted the attention calculation calculation method of the FLAT [28].Given input token sequence S, we first express the position of each token by indicating its head and tail. For latent words, the head and the tail denote the position index of the word's first and last characters in the original character sequence. For characters, their head and tail are identical. Therefore the flat-lattice structure can be defined as:

$$\{(s_1, head_1, tail_1), (s_2, head_2, tail_2), \dots, (s_{n+m}, head_{n+m}, tail_{n+m})\} \tag{4}$$

where $head_i$ is the head and $tail_i$ is the tail of the i^{th} token. With the help of flat-lattice structure, four kinds of relative distance between the i^{th} token and j^{th} token can be calculated as follows:

$$d_{ij}^{hh} = head_i - head_j \tag{5}$$

$$d_{ij}^{ht} = head_i - tail_j \tag{6}$$

$$d_{ij}^{th} = tail_i - head_j \tag{7}$$

$$d_{ij}^{tt} = tail_i - tail_j \tag{8}$$

where $d_{ij}^{hh}, d_{ij}^{ht}, d_{ij}^{th}, d_{ij}^{tt}$ represent the distance among heads, between head and tail and among tails, respectively. In order to considerate all these four kinds of positional relations, the relative position encoding of span R_{ij} is a non-linear transformation of four distances:

$$R_{ij} = RELU(W_r(\mathbf{P}_{d_{ij}^{hh}} \oplus \mathbf{P}_{d_{ij}^{ht}} \oplus \mathbf{P}_{d_{ij}^{th}} \oplus \mathbf{P}_{d_{ij}^{tt}})) \tag{9}$$

where W_r is a trainable parameters, $\oplus$ is the concatenation operator and $\mathbf{P}_d$ is calculated in the same way as Vaswani et al. [47]:

$$\mathbf{P}_d^{2k} = \sin(\frac{d}{10000^{2k/d_{model}}}) \tag{10}$$

$$\mathbf{P}_d^{2k+1} = \cos(\frac{d}{10000^{2k/d_{model}}}) \tag{11}$$

where d is $d_{ij}^{hh}, d_{ij}^{ht}, d_{ij}^{th}$ or d_{ij}^{tt}, d_{model} is the dimension of position encoding and k is its index. Afterwards the unmasked attention score is computed by a variant of self-attention [7]:

$$\mathbf{A}_{ij}^* = \mathbf{W}_q^\top s_i^\top s_j \mathbf{W}_k + \mathbf{W}_q^\top s_i^\top R_{ij} \mathbf{W}_{kR} + \mathbf{u}^\top s_j \mathbf{W}_k + \mathbf{v}^\top R_{ij} \mathbf{W}_{kR} \tag{12}$$

where $\mathbf{W}_q$, $\mathbf{W}_k$, $\mathbf{W}_{kR}$ and $\mathbf{u}, \mathbf{v}$ are parameters to learn. For simplicity, the attention scores computation can be denoted as a function $\mathbf{A}_{ij}^* = f(Q_i, K_j)$ that transforms query value Q_i and key value K_j to the attention score. Therefore the final masked attention score $A_{i,j}$ can be computed as:

$$A_{i,j} = \begin{cases} f(Q_i, K_j) & s_j \in \mathcal{N}_i \\ 10^{-15} & s_j \notin \mathcal{N}_i \end{cases} \tag{13}$$

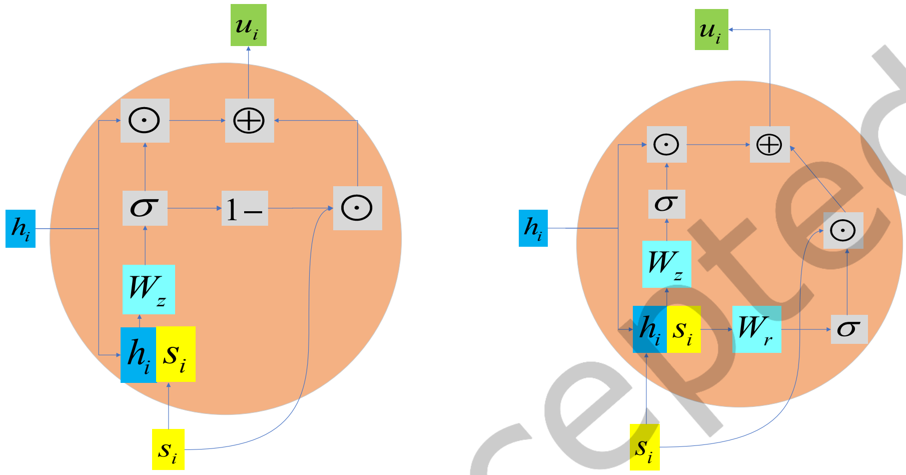
$$[Q_i, K_i, V_i] = s_i[W_q.W_k, W_v] \tag{14}$$

where W_q. W_k, W_v are trainable parameters. After getting the final attention score $A_{i,j}$, the final hidden state of token i^{th} can be obtained as below:

$$h_i = softmax(A_i)V \tag{15}$$

3.4 Gated Information Controller

The relations incorporated in the masked self-attention layer may bring misleading information. Therefore, two types of gated information controller, as shown in Fig. 3, are proposed to counterbalance the negative effect. As aforementioned, s_i and h_i represent the raw feature from the Token Encoding Layer and the hidden state from the masked self-attention layer, respectively. Therefore their mixed representation u_i can be computed in the following two different ways.



(a) Single Gated Approach

(b) Double Gated Approach

Fig. 3. The Gated Information Controller

The first approach, as shown in Fig. 3(a), is a single-gated approach. It applies one single *reset* gate r_i to evaluate the proportion of s_i and h_i as below:

$$r_i = \sigma(W_r \cdot [s_i, h_i]) \tag{16}$$

where W_r is the parameters to learn and σ is *sigmoid* function. Afterwards we use

$$u_i = (\mathbf{1} - r_i) \circ h_i + r_i \circ s_i \tag{17}$$

to control their combination, where u_i is the output of the gate mechanism information controller corresponding to input s_i, $\mathbf{1}$ is a 1-vector with its dimension matching r_i and $\circ$ the element-wise multiplication operation.

The second approach, as shown in Fig. 3(b), is a double-gated approach. It is equipped with an extra *update* gate z_i. It is calculated as:

$$z_i = \sigma(W_z \cdot [s_i, h_i]) \tag{18}$$

where W_z is trainable parameters. The outcome u_i is:

$$u_i = z_i \circ h_i + r_i \circ s_i \tag{19}$$

With the aid of Eq. (19), two sources of information s_i and h_i are controlled with two gates relevant r_i and z_i, respectively. Compared with single gated mechanism, it can generate more flexible and richer representations by getting rid of the restriction $z_i + r_i = \mathbf{1}$. However, owing to the increase of trainable parameters, it requires larger data volume for optimization.

3.5 Fully Connected Self-attention

After the gated mechanism controller, a fully connected multi-head self-attention layer is used to capture the long-range global information and to incorporate the lexical features from latent words into characters. Here we utilized the same attention mechanism $f(Q_i, K_j)$ as what in previous masked self-attention layer:

$$u'_i = softmax(A'_i)V' \tag{20}$$

$$A'_{i,j} = f(Q'_i, K'_j) \tag{21}$$

$$[Q'_i, K'_i, V'_i] = u_i[W'_q, W'_k, W'_v] \tag{22}$$

where W'_q, W'_k, W'_v are parameters to learn, A'_i represents the attention score corresponding to token s_i. So far, we have obtained the final abstract representation u'_i for each token s_i.

3.6 Decoding and Training

For the decoding process, we retain only the representation of characters and leave out the latent words, then a feed-forward neural network W_o, b_o is used to transform its dimension to the output space by $o_i = W_o \cdot u'_i + b_o$. Afterwards a standard CRF [23] is applied to predict the label $\hat{y}_i$:

$$\hat{y}_i = \arg\max_{y_i \in \mathcal{Y}} \frac{exp(W_{crf} \cdot o_i + b_{crf})}{\sum_{y_{i-1}y_i} exp(W_{crf} \cdot o_i + b_{crf})} \tag{23}$$

where $\mathcal{Y}$ is the set of all labels, W_{crf} and b_{crf} are trainable parameters.

For the training, we optimize the model by exploiting negative log-likelihood as the loss function. Given a set of training examples $\{X_i, Y_i\}|_{i=1}^{M}$, the loss can be defined as:

$$L = -\sum_{i=1}^{K} \log P(Y_i|X_i) \tag{24}$$

4 EXPERIMENTAL STUDY

4.1 Experiment Configuration

4.1.1 Dabwtasets. To validate the effectiveness of the proposed framework, four widely used Chinese NER datasets are employed in the experimental study, including **Ontonotes 4.0** [51], **MSRA** [25], **Resume** [60], and **Weibo** [40]. Among them, **Resume** and **Weibo** are both collected from Chinese online resources. **Resume** is built from resumes on Sina Finance[2]. **Weibo** is constructed with the messages drawn from the Chinese social media Sina Weibo[3]. **MSRA** and **Ontonotes** are from the news domain. Among them, **Resume** is annotated with 8 types of named entities, while **Weibio** and **Ontonotes 4.0** contain 4 kinds of traditional named entities, i.e., PER, ORG, GPE, LOC. However, **Weibo** differentiates the specific names from current names. **MSRA** only contains LOC, ORG and PER. The statistics of datasets are shown in Table 1.

The proposed framework aims at incorporating character adjacency relation, token dependency relation and character co-occurrence relation among words into the NER task. The statistic of character adjacency relation is relatively simple, as such here we present the richness of the latter two relations in each dataset. Thus we count the average number of dependency relations avg_{dep} and the average number of latent words avg_{lex} of each sentence. Since these two numbers both depend on the length of the sentence, we define the density of

[2] https://finance.sina.com.cn/stock/

[3] https://www.weibo.com

Table 1. The Statistics of Datasets

Dataset	Train	Dev	Test	Entity Types
Ontonotes 4.0	15778	4301	4346	4
Weibo	1350	270	270	4
MSRA	45000	-	3442	3
Resume	3821	463	477	8

dependency relations and density of latent words matched with lexicon as Eq. (25) and Eq. (26), respectively.

$$Density_{dep} = \frac{avg_{dep}}{avg_{len}} \times 100\% \tag{25}$$

$$Density_{lex} = \frac{avg_{lex}}{avg_{len}} \times 100\% \tag{26}$$

where avg_{len} represents the average length of sentence.

As shown in Table 2, the **Ontonotes 4.0**, **Weibo** and **MSRA** datasets are roughly at the same level while **Resume** dataset apparently differs from the others. Its dependency relation is relatively sparser than the others. On the contrary, the latent word is much richer than other datasets.

Table 2. The Characteristic of Each Datasets

	Characteristic	Ontonotes 4.0	Weibo	MSRA	Resume
Train	avg_{dep}	19.89	34.25	29.56	16.21
	avg_{lex}	13.91	21.34	23.37	25.28
	avg_{len}	30.54	54.45	47.84	32.48
	$Density_{dep}$	65.13%	62.90%	61.79%	**49.91%**
	$Density_{lex}$	45.55%	39.19%	41.20%	**77.83%**
Dev	avg_{dep}	28.76	33.60	-	15.01
	avg_{lex}	23.86	21.46	-	23.43
	avg_{len}	46.62	53.41	-	30
	$Density_{dep}$	61.69%	62.91%	-	**50.03%**
	$Density_{lex}$	51.18%	40.17%	-	**78.1%**
Test	avg_{dep}	29.42	34.36	30.62	15.96
	avg_{lex}	24.59	22.25	24.57	24.17
	avg_{len}	47.86	54.90	49.48	31.66
	$Density_{dep}$	61.47%	62.59%	61.87%	**50.41%**
	$Density_{lex}$	51.37%	40.50%	49.65%	**76.34%**

4.1.2 Implementation. Lexicon and word embedding are chosen identically for four datasets from the work of Zhang and Yang [60]. The final lexicon contains 704.4k words. In particular, the number of two-character words and three-character words are 291.5k and 278.1k, respectively. Here in Table 3, we reported two characteristics of the lexicon used: 1) the lexicon coverage, 2) the proportion of wrongly matched words (words that matched from the lexicon but not exist in the vocabulary). Due to the lack of gold word segmentation results for four datasets, we utilized the public Toolkit fastNLP[4] to deal with the character-level data and construct the vocabulary of each dataset.

Table 3. The Characteristic of lexicon for each dataset

Dataset	uncovered words/vocabulary size	wrongly matched/matched
weibo	2435/8856=27.5%	8019/14438=55.5%
Ontonotes4.0	6037/35336 = 17.1%	37798/67095 = 56.3%
MSRA	15962/57189 = 27.91%	70632/111290 = 63.47%
Resume	1844/6255 = 29.48%	7114/11523 = 61.74%
Overall	25.1%	61.08%

From Table 3, we can tell that about a quarter of words were uncovered in the lexicon, and in all words matched from the lexicon, nearly 60% of words do not appear in the vocabulary of the dataset. This also corroborates the lexical matching problem that we mentioned in the introduction. For implementation details, we apply the Chinese BERT released by Cui et al. [6] for encoding characters. The Bert Layer and the word embedding are updated along with other parameters during tthe raining procedure. We use the public Toolkit fastNLP to gather the dependency relations from the original character sequence. The proposed model is optimized by mini-batch Stochastic Gradient Descent (SGD) with learning rate and batch size depending on the dataset. The hyper-parameters are adjusted according to the performance on the development set. Because the development set of **MSRA** dataset is not available, we divide the training set with a proportion of 20% as the validation set. The model that performs best on the development set is chosen to be tested. The details of the hyperparameters for each dataset are reported in Table 4.

Table 4. The Hyper-parameters for each datasets

Hyper-parameters	Ontonotes4.0	Weibo	MSRA	Resume
Epoch	100	80	50	50
head$_{\text{number}}$	6	6	6	6
head$_{\text{dim}}$	20	16	16	10
Learning rate	10^{-3}	5×10^{-4}	10^{-3}	8×10^{-4}
Batch	10	10	12	12
Relation type	adjacency char co-occur dependency	adjacency char co-occur -	adjacency char co-occur dependency	adjacency char co-occur dependency
Gate type	Double	Single	Double	Double

The $head_{number}$ represents the number of heads in multi-head self-attention mechanism, $head_{dim}$ is the dimension of the abstract features per head. **Gate type** is to describe the type of information controller used in experiment, either the single gated controller (Single) or the double gated controller (Double). **Relation type** refers to the types of relation we mask in Masked Self-Attention layer. Since the **Weibo** dataset is drawn from social media platform, the majority of sentence does not follow the syntactic rules. As such the quality of

[4] https://github.com/fastnlp/fastNLP

dependency parsing results is much lower, we then exclude the dependency relations in this dataset. On the other hand, **Weibo** dataset is also too small thus we choose single gated information controller to avoid redundant trainable parameters.

4.1.3 Evaluation Metrics. Following most previous works in NER task, we choose Precision (P), Recall (R) and F1-score ($F1$) as evaluation indicators, which are calculated as:

$$P = \frac{TP}{TP + FP} \tag{27}$$

$$R = \frac{TP}{TP + FN} \tag{28}$$

$$F1 = \frac{2PR}{P + R} \tag{29}$$

where TP, FP, TN, FN represents true positive, false positive, true negative, false negative.

4.1.4 Baselines. To evaluate the effectiveness of our proposed framework, we select several advanced methods that also introduce syntactic information, such as dependency relation, or external lexicon as baselines. Besides, applying the pre-trained model such as BERT [9] or ZEN [10] as embedding layer is very helpful for model's performance. For a fair comparison, all of the baselines chosen have employed BERT as their character embedding layer.

1. **Lattice-LSTM+BERT** [60] is a Chinese NER model that exploits lexical information in character sequence through gated recurrent cells to avoid word segmentation errors.
2. **LR-CNN+BERT** is a CNN-based method that incorporates lexicons using a rethinking mechanism.
3. **SA-NER** [39] is an attentive semantic augmentation module that regards one token's most similar words as the semantic information and integrates them with Gating Module and Augmentation Module.
4. **SLK-NER** [18] is a model that incorporates informative lexicon knowledge by developing a method to compute a weighted sum of word information as the additional feature.
5. **PLTE+BERT** [54] is an extension of transformer encoder that is tailored for Chinese NER by introducing a porous mechanism.
6. **AESINER** [38] is a word-based labeling model that enhances NER through syntactic information including part-of-speech (POS) tags, syntactic constituents and dependency relations.
7. **SoftLexicon+BERT** [34] is a simple but effective method that incorporates the latent words by concatenating the different latent word embeddings according to their appearance order to construct extra features.
8. **FLAT+BERT** [28] is a flat-lattice Transformer to incorporate lexicon information for Chinese NER. We choose the version that is fine-tuned on BERT.

For a clearer illustration, the additional information used in different baselines are summarized in Table 5. The lexicon used in all baselines and in our work is identical. † denotes a word-based method that requires word sequence as input. We do not use the type and the direction of dependency relations. Besides, it is worth noting that the dependency relation will be discarded if its governor or its dependents do not exist in the lexicon used, thus no extra latent words are introduced.

4.2 Result and Analysis

4.2.1 Overall Performance. The overall experiment results are shown in Table 6 and Table 7. All of the experimental results for baselines are from published papers. Since the **SA-NER** is specially designed for social media data, it only conducts experiments on **Weibo** dataset.

Table 5. The additional information incorporated in different baseline models and our model.

Model	Additional Information
Lattice-LSTM+BERT [60]	Lexicon
LR-CNN+BERT [34]	Lexicon
SA-NER [39]	Lexicon
SLK-NER [18]	Lexicon
PLTE+BERT [54]	Lexicon
AESINER† [38]	Dependency, Word Segmentation, POS tag, Constituency
SoftLexicon+BERT [34]	Lexicon
FLAT+BERT [28]	Lexicon
Ours	Lexicon, Existence of dependency relation

Table 6. The Overall Experiment Results on Ontonotes4.0 and Weibo

Datset Method	**Ontonotes4.0**			**Weibo**		
	Precision	Recall	F1	Precision	Recall	F1
Lattice-LSTM+BERT	79.79	79.41	79.60	61.08	47.22	53.26
LR-CNN+BERT	79.41	80.32	79.86	64.11	67.77	65.89
SA-NER	-	-	-	-	-	69.80
SLK-NER	77.90	82.20	80.20	61.80	66.30	64.00
PLTE+BERT	79.62	81.82	80.60	72.00	66.67	69.23
AESINER	-	-	81.18	-	-	69.78
SoftLexicon+BERT	83.41	82.21	82.81	70.94	67.02	70.50
FLAT+BERT	-	-	81.82	-	-	68.55
Ours w/o dependency	81.94	84.06	82.89	71.00	73.21	**72.08**
Ours	82.57	83.99	**83.28**	72.82	66.02	69.62

As shown in Table 6 and Table 7, our proposed framework has achieved the state-of-the-art performance on **Ontonotes**, **Weibo** and **MSRA** datasets. In particular, the significant improvements have been attained on two more difficult datasets **Ontonotes** and **Weibo** in terms of $F1$ score. Our framework brings improvements because we investigate three kinds of crucial relations to locally contextualize the latent word embedding and to strengthen the connections between those strongly related tokens via the proposed masked attention layer. The Gated Information Controller successfully filter a portion of harmful and misleading information introduced. Furthermore, modeling different relations indifferently also takes effect for denoising. Among four datasets, our model's performance on **Resume** matches the SOTA but is still relatively inferior, it is partially because that the dependency relations in **Resume** is sparser than others so that our method is not able to retrieve sufficient useful information.

Besides, we can also observe how the dependency relations influence the model's performance from Table 6 and Table 7. The reason why we studied dependency relations solely is that we can not guarantee the dependency results are absolutely accurate, unlike other two relations. It is found that the dependency relations do help the model perform better in **Onto 4.0**, **MSRA** and **Resume** datasets, and the improvement in **MSRA** data is the most significant. In contrast, there is a significant decrease on the **Weibo** dataset after the introduction of

Table 7. The Overall Experiment Results on MSRA and Resume

Datset	MSRA			Resume		
Method	Precision	Recall	F1	Precision	Recall	F1
Lattice-LSTM+BERT	93.99	92.86	93.42	95.79	95.03	95.41
LR-CNN+BERT	94.68	94.03	94.35	95.68	96.44	96.06
SLK-NER	-	-	-	95.20	96.40	95.80
PLTE+BERT	94.91	94.15	94.53	96.16	96.75	96.45
AESINER	-	-	-	-	-	**96.62**
SoftLexicon+BERT	95.75	95.10	95.42	96.08	96.13	96.11
FLAT+BERT	-	-	96.09	-	-	95.68
Ours w/o dependency	95.89	95.35	95.62	95.96	96.19	96.08
Ours	96.08	96.18	**96.13**	96.01	96.50	96.36

dependencies. This is due to the fact that the utterances in the **weibo** dataset are collected from social media platforms and do not follow the syntactic rules strictly, which in turn may lead to a poor quality of dependency analysis.

4.2.2 Cross Validation. To validate the improvement brought by our proposed model, we have also conducted 5-fold cross validation on each dataset. Table 8 shows the means and the variances of three metric on each dataset. From Table 8 we can verify that our approach do perform well on two more difficult dataset **Weibo** and **Ontonotes4.0**.

Table 8. The 5-fold Cross Validation Results

Dataset	**Precision**	**Recall**	**F1**
Weibo	70.80±1.28	74.33±3.59	72.49±2.17
Ontonotes4.0	85.51±0.83	87.59±0.81	86.75±0.92
MSRA	96.08±0.4	95.70±0.3	95.89±0.1
Resume	95.94±0.25	96.24±0.35	95.94±0.25

4.2.3 Ablation Studies. In order to verify the effectiveness of each component proposed, we have conducted following sets of ablation experiments on all four datasets. The ablation experiments are set as follows:

- **FC**: in this case, only the fully connected self-attention layer is left between the token encoding layer and the decoding layer. We can consider it as the gate for Token Relation Aware Self Attention is totally closed.
- **2FC**: in this case, we stack one more fully connected self-attention layer on the basis of **FC**.
- **MFC**: in this case, we employ the mask strategy in Section 3.4 in the first layer of **2FC**. We can consider it as the gate for Token Encoding Layer is totally closed.
- **MSGFC**: in this case, we apply the single gated mechanism information controller.
- **MDGFC**: in this case, we apply the double gated mechanism information controller.

For all groups of ablation experiments, we maintain the same hyperparameters and the experiment settings (except **gate type**) that makes the model reach its best $F1$ score, as before, the dependency relations is also left out for all ablation experiment on **Weibo** dataset. Our considerations for designing these ablation experiments are:

(1) The improvement may simply due to the stack of the self-attention layers but not the strategies. So we set **FC** and **2FC** to explore the impact of stacking self-attention layer on the experimental results.
(2) We set **2FC** and **MFC** to investigate whether using solely mask strategy can improve the model's performance.
(3) We set the **MFC**, **MSGFC** and **MDGFC** to validate the effectiveness of the gated information controller mechanism and to explore how the features retrieved by Token Relation Aware Masked Self-Attention can be trusted.

Table 9. The Ablation Experimental Results

Dataset	**Metrics**	**FC**	**2FC**	**MFC**	**MSGFC**	**MDGFC**
Onto4.0	P	81.12	80.62	80.52	81.64	**82.57**
	R	83.98	84.92	84.30	**84.38**	83.99
	F1	82.52	82.71	82.37	82.99	**83.28**
Weibo	P	70.57	**71.76**	69.25	71.00	69.44
	R	64.35	67.46	70.57	**73.21**	73.92
	F1	65.21	69.54	69.90	**72.08**	71.61
MSRA	P	94.91	95.36	95.19	95.87	**96.08**
	R	95.58	95.13	95.68	95.73	**96.18**
	F1	95.02	95.52	95.39	95.80	**96.13**
Resume	P	94.34	93.82	95.55	95.97	**96.20**
	R	95.03	84.11	96.20	96.44	**96.50**
	F1	94.68	93.97	95.87	96.20	**96.36**

The ablation experiment results are shown in Table 9, where **MDGFC** has achieved the highest $F1$ score on **Ontonotes 4.0**, **MSRA** and **Resume** dataset and **MSGFC** has achieved the best performance on **Weibo** dataset, which indicates that every component of our framework is essential and indispensable. The analysis of our ablation studies are as follows:

(1) Simply stacking the fully connected self-attention layer does lead to the improvement on **Ontonotes4.0**, **Weibo** and **MSRA** datasets. The slight decline on **Resume** probably because it is the simplest dataset from the perspective of characteristic and volume of the data. One single fully connected self-attention layer **FC** can already recognize most of the entities from the textual sequence of **Resume**, redundant neurons that have similar functions may hurt the performance.
(2) By comparing the experimental result of **2FC** and **MFC**, it is found that the performance on **Weibo**, **Resume** improves after adding mask strategy. But on **Ontonotes 4.0** and **MSRA**, the opposite phenomenon is detected. For the first two datasets, the results of the dependency analysis in the **Weibo** dataset were not used, while the dependency relationships in the Resume dataset were very sparse and the character co-occurrence relationships were very rich, so the performance of the model on these two datasets increased rather than decreased. In the other two datasets, the dependency relations are dominated. As such in this case, the performance naturally deteriorates because the model lacks the ability to determine whether the

dependencies are plausible. Therefore this phenomenon maybe due to the poor quality of the dependence analysis results.

(3) It is also found that both of the **MSGFC** and **MDGFC** have reached higher $F1$ score than **MFC** on every dataset. The gated mechanism for information controllers play critical role as expected. To a certain extent, it counter-weighs the negative impact of introducing the improper and misleading relation information. Therefore it is very necessary to control the combination of the original token representation and the feature enriched with the external information, especially when the quality of the information is difficult to guarantee.

4.2.4 Impact of Density of Relations. The density of dependency relations and density of latent words are intrinsic properties of natural language. Since the two densities are not distinguished from each other in training process, we refer to sum of them as the density of relations. This characteristic may have impact on the performance of the proposed framework, which means that even in the same dataset, our approach may perform differently in response to utterances with different richness of relations. Therefore, to investigate its influence, we decide to train our framework that involves all of the three particular relations in mask attention layer on whole complete training set. Afterwards we classify the instances in the test set according to its density, finally we test our model on each of these sub-datasets and observe its $F1$ score. The way we divide the test set is to distribute it equally. For example, if we want to divide the test set into two sub-datasets, the first dataset will consist of half of the original data with lower density, while the second dataset consists of the rest data with higher density. For experiment settings, **Weibo** dataset is selected for representing the expressions from social media. Similarly, **Resume** is for Curriculum Vitae phrase and **Ontonotes 4.0** is for the news. **MSRA** belongs to the same domain as **Ontonotes 4.0**, as such here we only conduct experiment on **Ontonotes 4.0**. We divide their test sets into five sub-datasets and use the average density of each for indication.

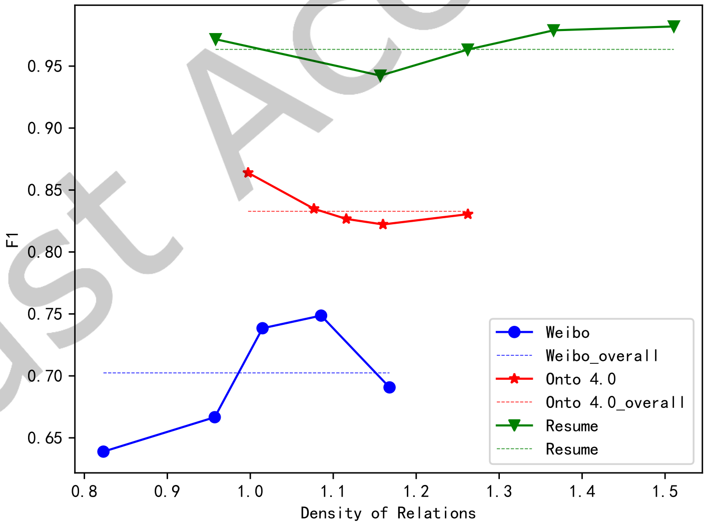
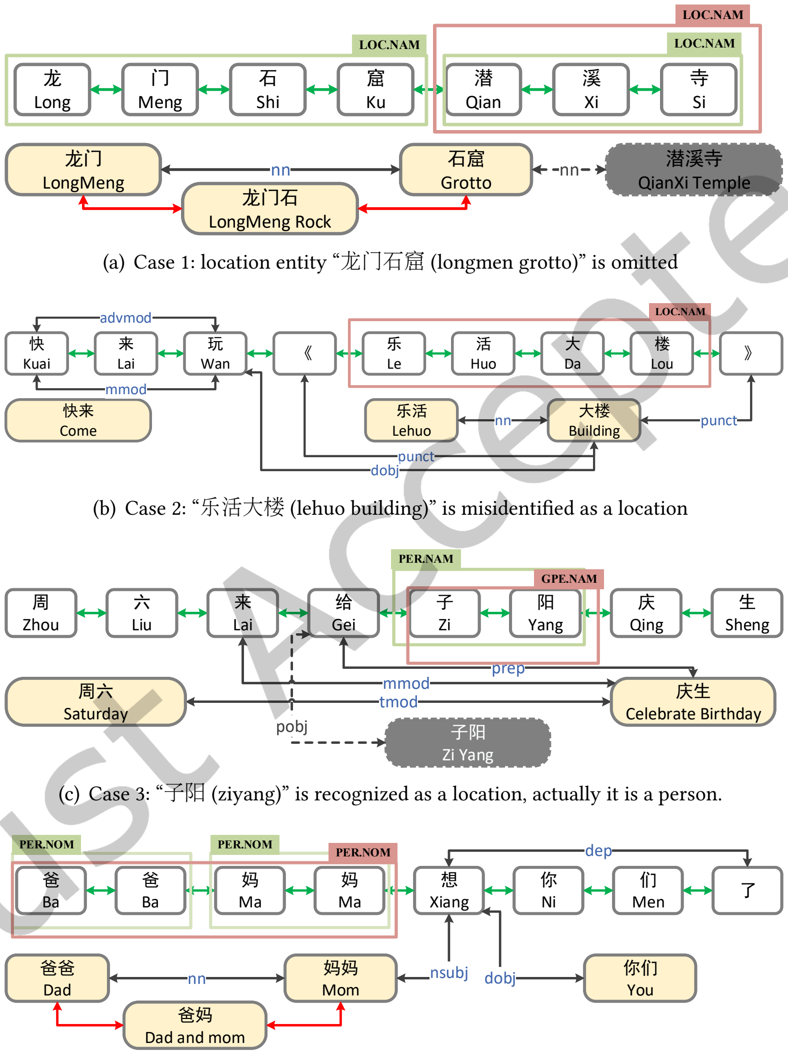


Fig. 4. The F1 score of our model on sentence with different relation density.

The experimental results are presented in Fig. 4. The horizontal axis indicates the density of relations, while the vertical axis indicates the evaluation results of the metrics. The dashed line indicates the performance of the model on the complete test set. Intuitively, language expressions in different domains show different variation tendency. Roughly, for **Weibo** and **Resume** dataset, the richer the relational information, the better the model's performance is. On the contrary, for dataset **Ontonotes 4.0** in news domain, the smaller the proportion of dependencies and character co-occurrence relations in the statement, the better the model performs.



(a) Case 1: location entity "龙门石窟 (longmen grotto)" is omitted

(b) Case 2: "乐活大楼 (lehuo building)" is misidentified as a location

(c) Case 3: "子阳 (ziyang)" is recognized as a location, actually it is a person.

(d) Case 4: "爸爸 (dad)" and "妈妈 (mom)" are recognized as one entity, actually they are two separate entities.

Fig. 5. Four cases that represent four common errors in Chinese NER: entities omitted, entity misidentified, entity type errors and entity boundary errors.

4.2.5 Case Study. In order to present how proposed approach brings the improvements for Chinese NER task intuitively, four cases are selected from **Weibo** test set and are shown in Fig. 5. While selecting the cases, we focused on four kinds of errors the occurred frequently in NER tasks:

(1) False negative, the entities are omitted.
(2) True negative, the non-entities are misidentified as entities.
(3) The entity type errors.
(4) The entity boundary errors.

For each case, we present the character sequence (white rectangles), the latent words matched from lexicon used (yellow rectangles), and the words that should be matched but not included in lexicon (gray rectangles). The black arrows represent dependency relations, green arrows represent adjacency relations and red arrows represent character co-occurrence relations. The gray blocks and the dashed lines represent the unmatched words and their corresponding relations, which are not introduced. For each case we present two kinds of labeling results: our model's (gold result, represented by green squares) and FLAT's (red squares).

(1) Case 1 is for named entity omitted errors, the location entity "龙门石窟 (longmen grotto)" is omitted.
(2) Case 2 is for named entity misidentified errors, the word "乐活大楼 (lehuo building)" is not a named entity while it is recognized as a location by FLAT model.
(3) Case 3 is for misjudgment of entity type, the "子阳 (ziyang)" is the name of a person but FLAT model recognized it as the name of location.
(4) Case 4 is for named entity boundary errors, the word "爸爸 (dad)" and "妈妈 (mom)" are two different named entities, and the FLAT model recognized it as one entire name entity.

These 4 cases demonstrate that our approach has the capability of mitigating the errors mentioned above to some extent. For instance, in case 1 we emphasize the relations among "龙门 (longmen)", "石窟 (grotto)" and "龙门石 (longmen Stone)" to help model recognize "龙门石窟 (longmen grotto)". In case 2, by integrating the relation between "玩 (play)" and "大楼 (building)", our model infers that "乐活大楼 (lehuo building)" is not a location. In case 3, the word "庆生 (celebrate birthday)" is highlighted (related with three tokens, the most) thus our model recognizes "子阳 (ziyang)" as a person but not the location. In case 4, our model has underlined the nn dependency relation (nn means modifying) between "爸爸 (dad)" and "妈妈 (mom)" so that we can recognize "爸爸妈妈 (dad and mom)" as one entire entity. The FLAT recognition errors may be due to their limitations mentioned in the Introduction. For example, in case 2, the meaning of the keyword "庆生 (celebrate birthday)" is not highlighted due to the fully connected model structure, leading to the identification of "子阳 (ziyang)" as location; in case 2, the important relation between "玩 (play)" and "大楼 (building)" is not captured, leading to recognize "乐活大楼 (lehuo building)" as a location.

5 CONCLUSION AND FUTURE WORK

Currently, introducing lexical features into character-based model has become a promising direction for Chinese NER. During the integration of character and word information, it is found that the latent word embedding introduced is still non-contextualized and there exist various kinds of relations among the tokens (characters and words). To fully utilize such information, in this research we investigated the importance of dependency relations among tokens, character co-occurrence relations, and character adjacency relations. Afterwards we proposed a token relation aware framework to incorporate these features into latent words and then into characters. In this framework, a masked self-attention mechanism is proposed to fully explore the relation. Furthermore, a gated information controller is developed to flexibly adjust the most useful information. Through the experimental results on four widely used datasets, the proposed framework proves its effectiveness in improving performance in Chinese NER. We also validate the necessity of each component of the proposed model by extensive ablation studies.

Although we have specially studied the Chinese NER task in this paper, it is believed that our proposed framework is equally applicable to other domain of NER task and even in other NLP tasks. The mask strategy proposed in this research maybe also applicable to some advanced neural networks such as graph neural networks, which has become a useful model in NLP domain.

ACKNOWLEDGMENTS

This work was partially supported by the State Key Laboratory of Software Development Environment of China (No. SKLSDE-2021ZX-16), and the National Natural Science Foundation of China (No. 61977002).

REFERENCES

[1] Adam L. Berger and John D. Lafferty. 2017. Information Retrieval as Statistical Translation. *SIGIR Forum* 51, 2 (2017), 219–226.

[2] Roi Blanco, Giuseppe Ottaviano, and Edgar Meij. 2015. Fast and Space-Efficient Entity Linking for Queries. In *Proceedings of the 8th ACM International Conference on Web Search and Data Mining*. 179–188.

[3] Xiaoling Cai, Shoubin Dong, and Jinlong Hu. 2019. A deep learning model incorporating part of speech and self-matching attention for named entity recognition of Chinese electronic medical records. *BMC Medical Informatics & Decision Making* 19-S, 2 (2019), 101–109.

[4] Miao Chen, Ganhui Lan, Fang Du, and Victor S. Lobanov. 2020. Joint Learning with Pre-trained Transformer on Named Entity Recognition and Relation Extraction Tasks for Clinical Analytics. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*. 234–242.

[5] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel P. Kuksa. 2011. Natural Language Processing (Almost) from Scratch. *Journal of Machine Learning Research* 12 (2011), 2493–2537.

[6] Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Ziqing Yang, Shijin Wang, and Guoping Hu. 2019. Pre-Training with Whole Word Masking for Chinese BERT. *CoRR* abs/1906.08101 (2019).

[7] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc Viet Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive Language Models beyond a Fixed-Length Context. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*. 2978–2988.

[8] Thanh Hai Dang, Hoang-Quynh Le, Trang M. Nguyen, and Sinh T. Vu. 2018. D3NER: Biomedical Named Entity Recognition using CRF-biLSTM Improved with Fine-tuned Embeddings of Various Linguistic Information. *Bioinformatics.* 34, 20 (2018), 3539–3546.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 4171–4186.

[10] Shizhe Diao, Jiaxin Bai, Yan Song, Tong Zhang, and Yonggang Wang. 2020. ZEN: Pre-training Chinese Text Encoder Enhanced by N-gram Representations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*. 4729–4740.

[11] Ruixue Ding, Pengjun Xie, Xiaoyan Zhang, Wei Lu, Linlin Li, and Luo Si. 2019. A Neural Multi-digraph Model for Chinese NER with Gazetteers. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*. 1462–1467.

[12] Cícero Nogueira dos Santos and Victor Guimarães. 2015. Boosting Named Entity Recognition with Neural Character Embeddings. In *Proceedings of the 5th Named Entity Workshop*. 25–33.

[13] Sean R Eddy. 1996. Hidden Markov Models. *Current Opinion in Structural Biology* 6, 3 (1996), 361–365.

[14] Chen Gong, Saihao Huang, Houquan Zhou, Zhenghua Li, Min Zhang, Zhefeng Wang, Baoxing Huai, and Nicholas Jing Yuan. 2021. An In-depth Study on Internal Structure of Chinese Words. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*. 5823–5833.

[15] Tao Gui, Ruotian Ma, Qi Zhang, Lujun Zhao, Yu-Gang Jiang, and Xuanjing Huang. 2019. CNN-Based Chinese NER with Lexicon Rethinking. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. 4982–4988.

[16] Tao Gui, Yicheng Zou, Qi Zhang, Minlong Peng, Jinlan Fu, Zhongyu Wei, and Xuanjing Huang. 2019. A Lexicon-Based Graph Neural Network for Chinese NER. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. 1040–1050.

[17] Qipeng Guo, Xipeng Qiu, Pengfei Liu, Yunfan Shao, Xiangyang Xue, and Zheng Zhang. 2019. Star-Transformer. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Human Language Technologies*. 1315–1325.

[18] Dou Hu and Lingwei Wei. 2020. SLK-NER: Exploiting Second-order Lexicon Knowledge for Chinese NER. In *Proceedings of the 32nd International Conference on Software Engineering and Knowledge Engineering*. 413–417.

[19] Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF Models for Sequence Tagging. *CoRR* abs/1508.01991 (2015).

[20] Zhanming Jie and Wei Lu. 2019. Dependency-Guided LSTM-CRF for Named Entity Recognition. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. 3860–3870.

[21] Yanliang Jin, Jinfei Xie, Weisi Guo, Can Luo, Dijia Wu, and Rui Wang. 2019. LSTM-CRF Neural Network With Gated Self Attention for Chinese NER. *IEEE Access* 7 (2019), 136694–136703.

[22] Saravanakumar Kandasamy and Aswani Kumar Cherukuri. 2020. Query expansion using named entity disambiguation for a question-answering system. *Concurrency and Computation: Practice and Experience* 32, 4 (2020).

[23] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the 18th International Conference on Machine Learning*. 282–289.

[24] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 260–270.

[25] Gina-Anne Levow. 2006. The third international Chinese language processing bakeoff: Word segmentation and named entity recognition. In *Proceedings of the 5th SIGHAN Workshop on Chinese Language Processing*. 108–117.

[26] Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2018. A Survey on Deep Learning for Named Entity Recognition. *CoRR* abs/1812.09449 (2018).

[27] Peng-Hsuan Li, Tsu-Jui Fu, and Wei-Yun Ma. 2020. Why Attention? Analyze BiLSTM Deficiency and Its Remedies in the Case of NER. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*. 8236–8244.

[28] Xiaonan Li, Hang Yan, Xipeng Qiu, and Xuanjing Huang. 2020. FLAT: Chinese NER Using Flat-Lattice Transformer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 6836–6842.

[29] Yuan Li, Guodong Du, Yan Xiang, Shaozi Li, Lei Ma, Dangguo Shao, Xiongbin Wang, and Haoyu Chen. 2020. Towards Chinese clinical named entity recognition by dynamic embedding using domain-specific knowledge. *Journal of Biomedical Informatics* 106 (2020), 103435.

[30] Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural Relation Extraction with Selective Attention over Instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. 2124–2133.

[31] Wei Liu, Xiyan Fu, Yue Zhang, and Wenming Xiao. 2021. Lexicon Enhanced Chinese Sequence Labeling Using BERT Adapter. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*. 5847–5858.

[32] Wei Liu, Tongge Xu, QingHua Xu, Jiayu Song, and Yueran Zu. 2019. An Encoding Strategy Based Word-Character LSTM for Chinese NER. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2379–2389.

[33] Zhangxun Liu, Conghui Zhu, and Tiejun Zhao. 2010. Chinese Named Entity Recognition with a Sequence Labeling Approach: Based on Characters, or Based on Words?. In *Proceedings of 6th International Conference on Intelligent Computing*. 634–640.

[34] Ruotian Ma, Minlong Peng, Qi Zhang, Zhongyu Wei, and Xuanjing Huang. 2020. Simplify the Usage of Lexicon in Chinese NER. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 5951–5960.

[35] Xindian Ma, Peng Zhang, Shuai Zhang, Nan Duan, Yuexian Hou, Ming Zhou, and Dawei Song. 2019. A Tensorized Transformer for Language Modeling. In *Proceedings of 2019 Annual Conference on Neural Information Processing Systems*. 2229–2239.

[36] Pedro Henrique Martins, Zita Marinho, and André F. T. Martins. 2019. Joint Learning of Named Entity Recognition and Entity Linking. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*. 190–196.

[37] Sewon Min, Victor Zhong, Richard Socher, and Caiming Xiong. 2018. Efficient and Robust Question Answering from Minimal Context over Documents. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. 1725–1735.

[38] Yuyang Nie, Yuanhe Tian, Yan Song, Xiang Ao, and Xiang Wan. 2020. Improving Named Entity Recognition with Attentive Ensemble of Syntactic Information. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*. 4231–4245.

[39] Yuyang Nie, Yuanhe Tian, Xiang Wan, Yan Song, and Bo Dai. 2020. Named Entity Recognition for Social Media Texts with Semantic Augmentation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. 1383–1391.

[40] Nanyun Peng and Mark Dredze. 2015. Named Entity Recognition for Chinese Social Media with Jointly Trained Embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. 548–554.

[41] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Human Language Technologies*. 2227–2237.

[42] Ying Qin, Taozheng Zhang, and Xiaojie Wang. 2008. Chinese Named Entity Recognition with new contextual features. In *Proceedings of the 4th International Conference on Natural Language Processing and Knowledge Engineering*. 1–6.

[43] Sunita Sarawagi and William W. Cohen. 2004. Semi-Markov Conditional Random Fields for Information Extraction. In *Proceedings of 2004 Annual Conference on Neural Information Processing Systems*. 1185–1192.

[44] Cijian Song, Yan Xiong, Wenchao Huang, and Lu Ma. 2020. Joint Self-Attention and Multi-Embeddings for Chinese Named Entity Recognition. In *Proceedings of 6th International Conference on Big Data Computing and Communications*. 76–80.

[45] Emma Strubell, Patrick Verga, David Belanger, and Andrew McCallum. 2017. Fast and Accurate Entity Recognition with Iterated Dilated Convolutions. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 2670–2680.

[46] Dianbo Sui, Yubo Chen, Kang Liu, Jun Zhao, and Shengping Liu. 2019. Leverage Lexical Knowledge for Chinese Named Entity Recognition via Collaborative Graph Network. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. 3828–3838.

[47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Proceedings of 2017 Annual Conference on Neural Information Processing Systems*. 5998–6008.

[48] Caiyu Wang, Hong Wang, Hui Zhuang, Wei Li, Shu Han, Hui Zhang, and Luhe Zhuang. 2020. Chinese medical named entity recognition based on multi-granularity semantic dictionary and multimodal tree. *Journal of Biomedical Informatics* 111 (2020), 103583.

[49] Qi Wang, Yangming Zhou, Tong Ruan, Daqi Gao, Yuhang Xia, and Ping He. 2019. Incorporating Dictionaries into Deep Neural Networks for the Chinese Clinical Named Entity Recognition. *Journal of Biomedical Informatics* 92 (2019).

[50] Rui Wang, Xin Xin, Wei Chang, Kun Ming, Biao Li, and Xin Fan. 2019. Chinese NER with Height-Limited Constituent Parsing. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*. 7160–7167.

[51] Ralph Weischedel, Sameer Pradhan, Lance Ramshaw, Martha Palmer, Nianwen Xue, Mitchell Marcus, Ann Taylor, Craig Greenberg, Eduard Hovy, Robert Belvin, and Ann Houston. 2011. Ontonotes Release 4.0. https://catalog.ldc.upenn.edu/docs/LDC2011T03/OntoNotes-Release-4.0.pdf.

[52] Shuang Wu, Xiaoning Song, and Zhen-Hua Feng. 2021. MECT: Multi-Metadata Embedding based Cross-Transformer for Chinese Named Entity Recognition. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*. 1529–1539.

[53] Mengge Xue, Weiming Cai, Jinsong Su, Linfeng Song, Yubin Ge, Yubao Liu, and Bin Wang. 2019. Neural Collective Entity Linking Based on Recurrent Random Walk Network Learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. 5327–5333.

[54] Mengge Xue, Bowen Yu, Tingwen Liu, Yue Zhang, Erli Meng, and Bin Wang. 2020. Porous Lattice Transformer Encoder for Chinese NER. In *Proceedings of the 28th International Conference on Computational Linguistics*. 3831–3841.

[55] Chengxi Yan, Qi Su, and Jun Wang. 2020. MoGCN: Mixture of Gated Convolutional Neural Network for Named Entity Recognition of Chinese Historical Texts. *IEEE Access* 8 (2020), 181629–181639.

[56] Hang Yan, Bocao Deng, Xiaonan Li, and Xipeng Qiu. 2019. TENER: Adapting Transformer Encoder for Named Entity Recognition. *CoRR* abs/1911.04474 (2019).

[57] Shuo Yan, Jianping Chai, and Liyun Wu. 2020. Bidirectional GRU with Multi-Head Attention for Chinese NER. In *Proceedings of 5th IEEE Information Technology and Mechatronics Engineering Conference*. 1160–1164.

[58] Naixin Zhang, Feng Li, Guangluan Xu, Wenkai Zhang, and Hongfeng Yu. 2019. Chinese NER Using Dynamic Meta-Embeddings. *IEEE Access* 7 (2019), 64450–64459.

[59] Naixin Zhang, Guangluan Xu, Zequn Zhang, and Feng Li. 2019. MIFM: Multi-Granularity Information Fusion Model for Chinese Named Entity Recognition. *IEEE Access* 7 (2019), 181648–181655.

[60] Yue Zhang and Jie Yang. 2018. Chinese NER Using Lattice LSTM. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. 1554–1564.

[61] Dandan Zhao, Jingxiang Cao, Degen Huang, Jiana Meng, and Pan Zhang. 2021. Dual Neural Network Fusion Model for Chinese Named Entity Recognition. *International Journal of Computational Intelligence Systems* 14, 1 (2021), 471–481.

[62] Yuying Zhu and Guoxin Wang. 2019. CAN-NER: Convolutional Attention Network for Chinese Named Entity Recognition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 3384–3393.