Chinese NER with Character Convolution Boundary Attention

Anonymous ACL submission

Abstract

We explore the boundary attention models for character-level Chinese NER. We test the standard transformer model, as well as a novel variant in which the encoder block combines information from the nearby global attention of characters using convolutions. The convolutions are activated by gate to represent boundaries. The boundaries are added into encode in forward to produce entity boundaries based on input sequence. We perform extensive experiments on four Chinese NER datasets. Our transformer variant consistently outperforms the standard transformer at the character-level and converges faster while learning more robust character-level alignments.

1 Introduction

Segments boundaries in a sentence are usually identified in NER. For boundary embedding, span representation is calculated by the concatenation of the start and end tokens' representations (Fu et al., 2021). To enumerate all possible text spans in a sentence, the concatenation of word representations of its startpoint and endpoint with a 20-dimensional embedding represent the span width (Li et al., 2021a) following previous work. While Li et al. (2021b) focus on named entity boundary detection, which is to detect the start and end boundaries of an entity mention in text, without predicting its type. With attention model, Yu et al. (2020) detect entity span with unified multimodal Transformer. Zhang et al. (2018) use adaptive co-attention network. Prior attention (Zhao et al., 2019; Zhuang et al., 2022) is able to improve the concentration of attention on the global context through an explicit selection of the most relevant segments. In addition, boundary smoothing applies the smoothing technique to entity boundaries, rather than labels (Zhu and Li, 2022). There are also inspiring tasks (Xu et al., 2021; Ma et al., 2022; Cao and Wang, 2022; Hong et al., 2022) about boundary representation that performer language structure in model.

In Chinese NER, a drawback of the purely character-based (He and Wang, 2008; Liu et al., 2010; Li et al., 2014) NER method is that the word information is not fully exploited. To attend Chinese word, the NER task is separated in two steps: Chinese Word segmentation(CWS) and processing (Yang et al., 2016; He and Sun, 2017b). The first step will output a large number of incorrect word segmentation results, which leads to unsatisfactory language processing. The new character-based partitioning methods (Liu et al., 2019; Sui et al., 2019; Gui et al., 2019; Ding et al., 2019) come back to stage and have been empirically proven to be effective. With consideration of word information, Zhang and Yang (2018); Peng et al. (2019); Li et al. (2020) incorporate word lexicons into the character-based NER model. The wrong word lexicons from vocab or segmentation still will be incorporated without considering the whole sentences for segmentation.

To address the issue, we perform Character Convolution Boundary Attention(CCBA) to comparing with CWS system. CCBA adopts a sequence to sequence model with Transformer. Specifically, the model employs convolutional layers to model boundary of each character. The states come from nearby attention of character and are activated by gate to represent boundaries. The boundaries are added into encode in forward to produce entity boundaries based on input sequence. Experimental results show our model outperforms on the performance. In summary, the main contributions of this paper include:

- We propose a simple but effective method for incorporating word boundaries into the character representations for Chinese NER.

- The proposed method is transferable to different sequence-labeling architectures and can be easily used in other nlp task.

2 Background

2.1 Transformer Attention Modules

Attention mechanism is first proposed in NMT (Bahdanau et al., 2015), fully used in Transformer (Vaswani et al., 2017) and reviewed in a survey of Transformer (Lin et al., 2021). It is hot spot and common methods (Dai et al., 2019; Radford et al., 2019; Devlin et al., 2019). It can be seen that the development of attention mechanism is very fast. This success is partly due to the self-attention component which enables the network to capture contextual information from the entire sequence (Su et al., 2018). In this paper, we implement our method based on Transformer encoder-decoder framework, where the encoder first maps the input sequence into a sequence of continuous representations and the decoder generates an output sequence from the continuous representations. The encoder and decoder are trained jointly to maximize the conditional probability of target sequence given a source sequence. Transformer adopts attention mechanism with Query-Key-Value (QKV) model. The scaled dot-product attention used by Transformer is given in Equation (1).

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{D_k}}\right)V, \quad (1)$$

where $Q \in \mathbb{R}^{N\times D_k}$,$K \in \mathbb{R}^{M\times D_k}$,$V \in \mathbb{R}^{N\times D_k}$; N and M denote the lengths of queries and keys (or values); D_k and D_v denote the dimensions of keys (or queries) and values; softmax is applied in a row-wise manner. The dot-products of queries and keys are divided by $\sqrt{D_k}$ to alleviate gradient vanishing problem of the softmax function.

2.2 Attention with Prior

Attention mechanism generally outputs an expected attended value as a weighted sum of vectors, where the weights are an attention distribution over the values. However, it is observed that for the trained Transformers the learned attention matrix is often very sparse across most data points. Therefore, it is possible to reduce computation complexity by incorporating structural bias to limit the number of query-key pairs that each query attends to. Under this limitation, we just compute the similarity score of the query-key pairs according to pre-defined patterns in Equation (2).

$$\text{Attention}(Q_f,K_f,V_f) = \text{softmax}\left(\frac{Q_pK_p^\top}{\sqrt{D_{k_p}}}\right)V_p \oplus \text{softmax}\left(\frac{Q_gK_g^\top}{\sqrt{D_{k_g}}}\right)V_g. \quad (2)$$

Where Q_g, K_g, V_g is calculated by the vector query value, key value, extraction value for global attention; Q_p, K_p, V_p is calculated by the vector query value, key value, extraction value for prior attention; Q_f, K_f, V_f is calculated by the vector query value, key value, extraction value for final attention; D_{k_g} is the dimension of K_g; D_{k_p} is the dimension of K_p.

2.3 Convolutional Transformer

To facilitate character-level interactions in the transformer, *convtransformer* (Gao et al., 2020) propose a modification of the standard architecture. In this architecture, they use the same decoder as the standard transformer, but they adapt each encoder block to include an additional sub-block. Inspired from Lee et al. (2017), it is applied to the input representations M, before applying self-attention. The sub-block consists of three 1D convolutional layers, C_w, with different context window sizes w. In order to maintain the temporal resolution of convolutions, the padding is set to $\lfloor \frac{w-1}{2} \rfloor$.

For all convolutional layers, they set the number of filters to be equal to the embedding dimension size d_{model}, which results in an output of equal dimension as the input M. Therefore, in contrast to Lee et al. (2017), who use max-pooling to compress the input character sequence into segments of characters, here they leave the resolution unchanged, for both transformer and convtransformer models. Finally, for additional flexibility, they add a residual connection (He et al., 2016) from the input to the output of the convolutional block.

3 Method

3.1 Convolution Boundary Attention

In convolutional attention structure(Figure 1),we apply convolutional layers C_2, using context window sizes of 2. The context window sizes aim to resemble boundary B_t between adjacent characters.

$$C_2(B_t) = W_t^b(A_t \oplus A_{t+1}), \quad (3)$$

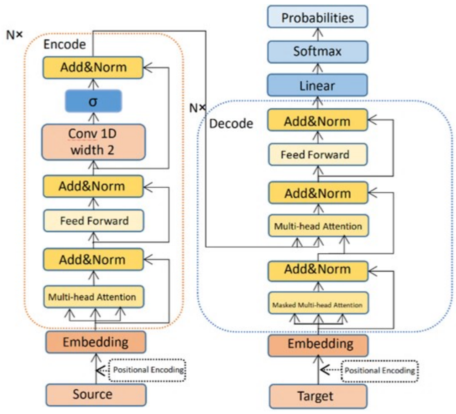

Figure 1: Convolutional attention structure.

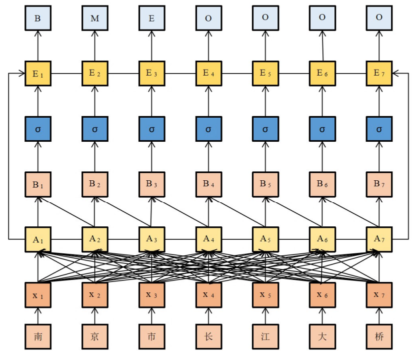

Figure 2: Attention model of character convolution boundary.

where $W_t^b \in \mathbb{R}^{d_{\text{model}} \times k \times n}$, k is kernel size, n is number of convolution. A_t is attention of character in time step t. $\oplus$ contacts nearby attention for convolution.

To compute the boundaries, the convolutional layers are activated with sigmoid σ and added with attention layers in encode, which fuses the representations:

$$Encode = A + \sigma(W^b(C_2(B_t))). \tag{4}$$

where $W^b \in \mathbb{R}^{m \times d_{\text{model}}}$, m is channels of input. A is attention of characters.

For all convolutional layers, we set the number of filters to be equal to the embedding dimension size d_{model}, which results in an output of equal dimension as the feed forward in encode.

3.2 Attention Model of Chinese NER

We design the attention mechanism model for Chinese NER. The global attention calculation is performed on the character vector to obtain the global weights and the convolutional attention calculation to obtain the boundary. The attention weights are computed separately for each character to form convolutional attention mechanism and combined with the global attention mechanism to input the model to obtain the results. Figure 2 is an example for details.

In Figure 2, the character sequences of ['南(South)', '京(Capital)', '市(City)', '长(Long)', '江(River)' , '大(Big)', '桥(Bridge)'], which are pretrained with unigram and bigram embeddings, result in character vector groups $(x_1, x_2, x_3, x_4, x_5, x_6, x_7)$ respectively. Global attention calculation of character vector groups results in a weight of $(A_1, A_2, A_3, A_4, A_5, A_6, A_7)$. Convolutional attention calculation results in a weight of $(B_1, B_2, B_3, B_4, B_5, B_6, B_7)$ and is activated by sigmoid σ and added global attention weights to form encode block $(E_1, E_2, E_3, E_4, E_5, E_6, E_7)$. The results are obtained with labels (B, M, E, O, O, O, O).

4 Experiment

4.1 Setup

Datasets. The model is evaluated on four Chinese NER datasets, including MSRA (Levow, 2006), OntoNotes (Weischedel et al., 2011), Resume NER (Zhang and Yang, 2018) and Weibo NER (Peng and Dredze, 2015; He and Sun, 2017a). Weibo NER is a social media domain dataset, which is drawn from Sina Weibo, while OntoNotes and MSRA datasets are in the news domain. Resume NER dataset consists of resumes of senior executives, which is annotated by (Zhang and Yang, 2018).

Evaluation. We use P, R and F1 in average to evaluate our performance on MSRA, OntoNotes and Resume datasets comparing with other methods. We used F1 in average to evaluate our performance on the NE, NM and Overall of Weibo dataset comparing with other methods. Transformer is baseline model to evaluate the function depending on our method. TENER and Flat evaluate the function from method in cooperation.

Model settings. For model, we adopted similar settings as TENER(`https://github.com/fastnlp/TENER`) and Flat(`https://github.com/LeeSureman/Flat-Lattice-Transformer`). We download the specified pretrained unigram and bigram embeddings for Chinese task. Most implementation details include character and word embedding

Models	P	R	F1
Chen et al. (2006)	91.22	81.71	86.20
Zhang et al. (2006)*	92.20	90.18	91.18
Zhou et al. (2013)	91.86	88.75	90.28
Lu et al. (2016)	-	-	87.94
Dong et al. (2016)	91.28	90.62	90.95
Ma et al. (2020)* †	**94.63**	**92.70**	**93.66**
Transformer(Baseline)	90.23	90.52	90.32
CCBA(ours)	92.26	90.68	91.46
TENER(Yan et al., 2019)	92.97	91.96	92.46
CCBA+TENER	93.00	92.61	92.80
Flat(Li et al., 2020)* †	92.46	93.77	93.11
CCBA+Flat	**93.06**	**94.13**	**93.59**

Table 1: Performance on MSRA.

Models	P	R	F1
Yang et al. (2016)	65.59	71.84	68.57
Yang et al. (2016)*†	72.98	**80.15**	**76.40**
Che et al. (2013)*	**77.71**	72.51	75.02
Wang et al. (2013)*	76.43	72.32	74.32
Ma et al. (2020)* †	77.13	75.22	76.16
Transformer(Baseline)	73.60	73.81	73.69
CCBA(Ours)	75.51	75.90	75.70
TENER(Yan et al., 2019)	75.97	77.29	76.63
CCBA+TENER	**76.41**	**77.31**	**76.85**
Flat(Li et al., 2020)* †	74.73	76.70	75.70
CCBA+Flat	75.06	77.21	76.12

Table 2: Performance on OntoNotes.

Models	P	R	F1
Zhang and Yang (2018)*	93.72	93.44	93.58
Zhu and Wang (2019)	94.07	94.42	94.24
Liu et al. (2019)*	93.66	93.31	93.48
Ding et al. (2019)	94.53	94.29	94.41
Ma et al. (2020)* †	**96.14**	**94.72**	**95.43**
Transformer(Baseline)	92.49	92.49	92.49
CCBA(Ours)	94.64	94.78	94.71
TENER(Yan et al., 2019)	94.79	94.97	94.88
CCBA+TENER	95.09	95.03	95.06
Flat(Li et al., 2020)* †	95.71	**95.77**	95.74
CCBA+Flat	**96.50**	95.33	**95.91**

Table 3: Performance on Resume.

Models	NE	NM	Overall
Peng and Dredze (2015)	51.96	61.05	56.05
Peng and Dredze (2016)*	55.28	62.97	58.99
He and Sun (2017a)	50.60	59.32	54.82
He and Sun (2017b)*	54.50	62.17	58.23
Ma et al. (2020)* †	**58.12**	**64.20**	**59.81**
Transformer(Baseline)	52.98	60.59	56.59
CCBA(Ours)	53.08	61.48	57.85
TENER(Yan et al., 2019)	55.06	63.72	58.82
CCBA+TENER	56.20	64.31	59.28
Flat(Li et al., 2020)* †	61.67	**65.27**	63.42
CCBA+Flat	**65.77**	62.05	**63.80**

Table 4: Performance on Weibo. NE, NM and Overall denote F1 scores for named entities, nominal entities (excluding named entities) and both, respectively.

sizes, dropout, embedding initialization, and transformer layer number. The convolutions is set as $((d_{\text{model}}, 2),) * 20$, d_{model} is set as 512.

4.2 Effectiveness Study

We conduct experiments on the four datasets to further verify the effectiveness of model in combination with pre-trained model. The results are shown in Tables 1−4. In these experiments, we use embedding encoders to obtain the character representations. Tables 1−4[1] show results on the MSRA, OntoNotes, Resume and Weibo datasets respectively against the compared baselines.

In Tables 1−4, compared methods include the best statistical models on these data set, which leveraged rich handcrafted features (Chen et al., 2006; Zhang et al., 2006; Zhou et al., 2013), character embedding features (Lu et al., 2016; Peng and Dredze, 2016), radical features (Dong et al., 2016), cross-domain data, semi-supervised data (He and Sun, 2017b) and incorporating word lexicons methods (Zhang and Yang, 2018; Peng et al., 2019; Li et al., 2020). The Transformer is base model to make clear whether the main improvement over the existing work is brought by CCBA. From the tables, we can see that the performance of the CCBA method is better than baseline methods on four datasets. The average performance of the CCBA method is near to SOTA on four datasets. The reason of cannot over SOTA may be the embedding in static state and depending on labels which may fail to recognize unnamed words like '江大桥(Daqiao Jiang)'. Comparing with TENER , we find that, CCBA+TENER have an improvement over TENER. Comparing with Flat, we find that, CCBA+Flat have an improvement over Flat. Those results show our method is transferable to different sequence-labeling architecture and improve the F1 in Chinese NER.

The proposed method (Li et al., 2020) employs a lattice-transformer and considers the multiple

[1]In Table 1−4, ∗ indicates that the model uses external labeled data for semi-supervised learning. † means that the model also uses discrete features.

Models	MSRA	OntoNotes	Resume	Weibo
CCBA	91.46	75.70	94.71	57.85
convtransformer	90.69	74.06	93.75	57.44
Prior attention	90.61	73.95	92.72	57.17

Table 5: An ablation study of the proposed model.

tokenizations. The real difference between our method and the proposed method should be discussed. However, they incorporate many wrong word lexicons without considering the whole sentences for segmentation. For example, in the sentence "南京市长江大桥(Nanjing Yangtze River Bridge)", they will incorporate wrong word '京市(Jing City)' without considering the whole sentence for segmentation. In our method, we consider the whole sentence for boundary attention. The smoothing technique to entity boundaries is better than hard word incorporation to show sentence structure and relationship between nearby characters.

For Chinese NER, the self-attention in Transformer is sparse and unbalanced of each character. We convolute nearby attention of character with context for boundary. It can be trained fast and reduce the parameters in model. During the boundary attention of sentence, we provide a soft way to locate the word boundary. It simplifies the model to learn structure from large data. With the additional attention of boundary, the model can fast and better learn the structure of sentence.

4.3 Ablation Study

To investigate the contribution of each component of our method, we conduct ablation experiments on all four datasets, as shown in table 5.

In the "*convtransformer*" experiment, we remove the boundary attention in CCBA and add convolutional layers like Gao et al. (2020). We consider the different convolution layers which is expediently encoded in the input. The input enhances with relative character and balances the length or information of each segmentation in sentence. We apply convolutional layers C_2, using context window sizes of 2. It is applied to the input representations M, which fuses the representations:

$$\mathrm{Conv}(M) = M + C_2, \tag{5}$$

The degradation in performance on all four datasets indicates the importance of convolution of nearby attention rather than character, and confirms the advantage of our method. The convolutional layers after attention layers can catch the context information in whole sentences to get soft boundary. The convolutional layers before attention layers are short of important wegthts of sentence.

In the "Prior attention" experiment, we remove the boundary attention in CCBA and add prior attention. In the model, we compute attention $\hat{Y}$ with prior attention A_2, using context window sizes of 2. The incorporation $\oplus$ in Equation (2) is shown in details in Equation (6) and Equation (7).

$$\hat{Y} = A \oplus A_2 = (1 - p_l) * A + p_l * A_2, \tag{6}$$

Where $p_l \in [0, 1]$ is a calculated probability, which balances the probability of global attention and local attention.

$$p_l = \sigma(W_2(W_1 H_{dec} + b_1) + b_2). \tag{7}$$

Where H_{dec} represent the decoder hidden state at timestep t and d_{model} to denote the dimension of the hidden states; $W_1 \in \mathbb{R}^{d_{model} \times d_{model}}$ and $W_2 \in \mathbb{R}^{1 \times d_{model}}$ are learnable matrices, $b_1 \in \mathbb{R}^{d_{model}}$ and $b_2 \in \mathbb{R}^1$ are bias vectors, σ is the sigmoid function.

The degradation in performance on all four datasets indicates the importance of the convolutional attention, and confirms the advantage of our method. The convolutional attention comes from nearby attention while prior attention comes from nearby character vectors. The former can better catch the context information in whole sentences to get soft boundary.

4.4 Compatibility with BERT

We compare CCBA with BERT on four datasets. We download the specified pretrained BERT model provided by huggingface. We use bert-base-chinese (https://storage.googleapis.com/bert_models/2018_11_03/chinese_L-12_H-768_A-12.zip) for Chinese task. Most implementation details followed those of BERT-NER (https://github.com/lemonhu/NER-BERT-pytorch), including character and word embedding sizes, dropout, embedding initialization, and transformer layer number. In these experiments, we first use a BERT encoder to obtain the contextual representations of each sequence, and then concatenated them into the character representations. Results are shown in Table 6.

From the table, we can see that the CCBA method with BERT outperforms the BERT tagger on all four datasets. These results show that

Models	MSRA	OntoNotes	Resume	Weibo
Baseline+BERT	93.63	77.93	95.68	62.07
+CCBA	93.76	78.19	95.91	63.80
TENER+BERT	94.69	82.06	94.75	67.44
+CCBA	95.88	82.86	95.72	68.02
Flat+BERT	96.09	81.82	95.86	68.55
+CCBA	96.61	81.95	96.02	69.17

Table 6: Compatibility with BERT.

the CCBA method can be effectively combined with pre-trained model. Moreover, the results also verify the effectiveness of our method in utilizing lexicon information, which means it can complement the information obtained from the pre-trained model. We also find that, CCBA+TENER+BERT have an improvement over TENER+BERT and CCBA+Flat+BERT have an improvement over Flat+BERT. Those results show our method is transferable to different sequence-labeling architecture and improve the F1 in Chinese NER with pre-trained model.

5 Conclusion

In this work, we address the convolutional attention of character boundary in Chinese NER. We propose a novel method to model sentence structure with considering the sequence of characters in whole sentence, which reduces many wrong words incorporated into the character representations. We use boundary attention with convolution instead of CWS system to embed the word-lever information. Experimental studies show that our performances have an improvement of existing methods.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Shuyang Cao and Lu Wang. 2022. HIBRIDS: Attention with hierarchical biases for structure-aware long document summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 786–807, Dublin, Ireland. Association for Computational Linguistics.

Wanxiang Che, Mengqiu Wang, Christopher D Manning, and Ting Liu. 2013. Named entity recognition with bilingual constraints. In *NAACL*, pages 52–62.

Aitao Chen, Fuchun Peng, Roy Shan, and Gordon Sun. 2006. Chinese named entity recognition with conditional probabilistic models. In *SIGHAN Workshop on Chinese Language Processing*.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of ACL*, pages 2978–2988, Florence, Italy.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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 Annual Meeting of the Association for Computational Linguistics*, pages 1462–1467, Florence, Italy. Association for Computational Linguistics.

Chuanhai Dong, Jiajun Zhang, Chengqing Zong, Masanori Hattori, and Hui Di. 2016. Character-based lstm-crf with radical-level features for chinese named entity recognition. In *Natural Language Understanding and Intelligent Applications*, pages 239–250. Springer.

Jinlan Fu, Xuanjing Huang, and Pengfei Liu. 2021. SpanNER: Named entity re-/recognition as span prediction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7183–7195, Online. Association for Computational Linguistics.

Yingqiang Gao, Nikola I. Nikolov, Yuhuang Hu, and Richard H.R. Hahnloser. 2020. Character-level translation with self-attention. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1591–1604, Online. Association for Computational Linguistics.

Tao Gui, Yicheng Zou, Qi Zhang, Minlong Peng, Jinlan Fu, Zhongyu Wei, and Xuan-Jing 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 (EMNLP-IJCNLP)*, pages 1039–1049.

Hangfeng He and Xu Sun. 2017a. F-score driven max margin neural network for named entity recognition in chinese social media. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 713–718.

Hangfeng He and Xu Sun. 2017b. A unified model for cross-domain and semi-supervised named entity recognition in chinese social media. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Jingzhou He and Houfeng Wang. 2008. Chinese named entity recognition and word segmentation based on character. In *Proceedings of the Sixth SIGHAN Workshop on Chinese Language Processing*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Wu Hong, Zhuosheng Zhang, Jinyuan Wang, and Hai Zhao. 2022. Sentence-aware contrastive learning for open-domain passage retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1062–1074, Dublin, Ireland. Association for Computational Linguistics.

Jason Lee, Kyunghyun Cho, and Thomas Hofmann. 2017. Fully Character-level Neural Machine Translation without Explicit Segmentation. *Transactions of the Association for Computational Linguistics*, 5:365–378.

Gina-Anne Levow. 2006. The third international chinese language processing bakeoff: Word segmentation and named entity recognition. In *SIGHAN Workshop on Chinese Language Processing*, pages 108–117.

Fei Li, Zhichao Lin, Meishan Zhang, and Donghong Ji. 2021a. A span-based model for joint overlapped and discontinuous named entity recognition. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4814–4828, Online. Association for Computational Linguistics.

Haibo Li, Masato Hagiwara, Qi Li, and Heng Ji. 2014. Comparison of the impact of word segmentation on name tagging for chinese and japanese. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland. European Language Resources Association (ELRA).

Jing Li, Aixin Sun, and Yukun Ma. 2021b. Neural named entity boundary detection. *IEEE Transactions on Knowledge and Data Engineering*, 33(4):1790–1795.

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*, pages 6836–6842, Online. Association for Computational Linguistics.

Tianyang Lin, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. 2021. A survey of transformers. *CoRR*, abs/2106.04554.

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, Volume 1 (Long and Short Papers)*, pages 2379–2389, Minneapolis, Minnesota. Association for Computational Linguistics.

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 *Advanced intelligent computing theories and applications. With aspects of artificial intelligence*, pages 634–640. Springer.

Yanan Lu, Yue Zhang, and Dong-Hong Ji. 2016. Multi-prototype chinese character embedding. In *LREC*.

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*, pages 5951–5960, Online. Association for Computational Linguistics.

Xinbei Ma, Zhuosheng Zhang, and Hai Zhao. 2022. Structural characterization for dialogue disentanglement. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 285–297, Dublin, Ireland. Association for Computational Linguistics.

Minlong Peng, Ruotian Ma, Qi Zhang, and Xuanjing Huang. 2019. Simplify the usage of lexicon in chinese ner. *ArXiv*, abs/1908.05969.

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*, pages 548–554.

Nanyun Peng and Mark Dredze. 2016. Improving named entity recognition for chinese social media with word segmentation representation learning. In *ACL*, page 149.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Jinsong Su, Jiali Zeng, Deyi Xiong, Yang Liu, Mingxuan Wang, and Jun Xie. 2018. A hierarchy-to-sequence attentional neural machine translation model. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(3):623–632.

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 (EMNLP-IJCNLP)*, pages 3821–3831.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Mengqiu Wang, Wanxiang Che, and Christopher D Manning. 2013. Effective bilingual constraints for semi-supervised learning of named entity recognizers. In *AAAI*.

Ralph Weischedel, Sameer Pradhan, Lance Ramshaw, Martha Palmer, Nianwen Xue, Mitchell Marcus, Ann Taylor, Craig Greenberg, Eduard Hovy, Robert Belvin, et al. 2011. Ontonotes release 4.0. *LDC2011T03, Philadelphia, Penn.: Linguistic Data Consortium*.

Lu Xu, Zhanming Jie, Wei Lu, and Lidong Bing. 2021. Better feature integration for named entity recognition. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3457–3469, Online. Association for Computational Linguistics.

Hang Yan, Bocao Deng, Xiaonan Li, and Xipeng Qiu. 2019. Tener: Adapting transformer encoder for named entity recognition.

Jie Yang, Zhiyang Teng, Meishan Zhang, and Yue Zhang. 2016. Combining discrete and neural features for sequence labeling. In *CICLing*. Springer.

Jianfei Yu, Jing Jiang, Li Yang, and Rui Xia. 2020. Improving multimodal named entity recognition via entity span detection with unified multimodal transformer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3342–3352, Online. Association for Computational Linguistics.

Qi Zhang, Jinlan Fu, Xiaoyu Liu, and Xuanjing Huang. 2018. Adaptive co-attention network for named entity recognition in tweets. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press.

Suxiang Zhang, Ying Qin, Juan Wen, and Xiaojie Wang. 2006. Word segmentation and named entity recognition for sighan bakeoff3. In *SIGHAN Workshop on Chinese Language Processing*, pages 158–161.

Yue Zhang and Jie Yang. 2018. Chinese ner using lattice lstm. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL), 1554-1564.*

Guangxiang Zhao, Junyang Lin, Zhiyuan Zhang, Xuancheng Ren, Qi Su, and Xu Sun. 2019. Explicit sparse transformer: Concentrated attention through explicit selection.

Junsheng Zhou, Weiguang Qu, and Fen Zhang. 2013. Chinese named entity recognition via joint identification and categorization. *Chinese journal of electronics*, 22(2):225–230.

Enwei Zhu and Jinpeng Li. 2022. Boundary smoothing for named entity recognition. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7096–7108, Dublin, Ireland. Association for Computational Linguistics.

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, Volume 1 (Long and Short Papers)*, pages 3384–3393, Minneapolis, Minnesota. Association for Computational Linguistics.

Yimeng Zhuang, Jing Zhang, and Mei Tu. 2022. Long-range sequence modeling with predictable sparse attention. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 234–243, Dublin, Ireland. Association for Computational Linguistics.