Chinese Word Attention based on Valid Division of Sentence

Anonymous ACL submission

Abstract

Chinese word attention (CWA) with word-level information is very important for natural language processing. The purpose is how to attention words in a sentence. We first explore the valid divisions of a sentence by splitting word tools. We use BERT for character and word pre-training. Each character embedding with its word in one division is encoded in block local attention. We use attention with prior to assign attention weights to each splitting result, and finally combine the global attention mechanism to get the optimal recognition result in Chinese NER.

1 Introduction

Language equals speech plus structure, and without boundaries there is no structure. In contrast to English, Chinese is a sequence of characters. There is no separator between characters (Su et al., 2018; Li et al., 2014), so word boundaries cannot be directly displayed. However, word-level information is very important for natural language processing (Mao et al., 2008; Peng and Dredze, 2016b; Zhang and Yang, 2018). Different ways of defining words can lead to different word separation results. There are still some basic questions like "what is a word" and "a word is what" that are not answered. Research shows that even if one is a native Chinese speaker, the rate of agreement on words appearing in Chinese texts is only about 70%. Therefore, in a strict sense, automatic word separation is a problem that is not clearly defined.

Traditionally, for Chinese NER, Chinese Word segmentation(CWS) system is first performed (Yang et al., 2016; He and Sun, 2017b). However, the existing CWS output a large number of incorrect word separation results, which leads to unsatisfactory language processing. In contrast to word-based partitioning methods, character-based partitioning methods (He and Wang, 2008; Liu et al., 2010; Li et al., 2014; Liu et al., 2019; Sui et al., 2019; Gui et al., 2019; Ding et al., 2019) have been empirically proven to be effective. A drawback of the purely character-based NER method is that the word information is not fully exploited. With this consideration, word lexicons are incorporated into the character-based NER model (Zhang and Yang, 2018; Peng et al., 2019; Li et al., 2020). However, they incorporate many wrong word lexicons without considering the whole sentences for splitting.

To address the issue, we performer Chinese word Attention(CWA) to comparing with CWS system. We explore how the splitting information can be effectively used and propose an attention mechanism with uncertain splitting boundaries. In this work we present an alternative approach, including valid division of words and computing attention weights. By the splitting tool, we search all the possibilities of splitting words to form valid division, excluding non-word divisions of sentence. We use BERT for character and word pre-training. Each character embedding with its word in one division is encoded in block local attention. We assign attention weights to each splitting result by attention with prior, and finally combine the global attention mechanism to get the optimal recognition result.

2 Background

2.1 Transformer Attention Modules

Transformer adopts attention mechanism with Query-Key-Value (QKV) model. The scaled dot-product attention used by Transformer is given in Equation (1).

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

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 Block Sparse Transformer

Self-attention plays an important role in Transformer. In the standard self-attention mechanism, every token needs to attend to all other tokens. 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.

Sparse attention (Parmar et al., 2018; Tay et al., 2020) inputs attention segments into several query blocks, each of which is associated with a local memory block. All the queries in a query block attend to only the keys in the corresponding memory block. To compute self-attention on the resulting sentences, we then partition the length into query blocks Q of length l_q, padding with zeroes if necessary. We partition the input tensor with positional encoding into rectangular query blocks contiguous in the original sentence. We generate one query block after another, ordering the blocks in order. Within each block, we generate individual positions.

2.3 Transformer 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. Traditionally, the distribution is generated from inputs, as depicted in Equation (2). As a generalized case, attention distribution can also come from other sources. Prior attention distribution can be a supplement or substitute for distribution generated from inputs. In most cases, the fusion of two attention distribution can be done by computing a weighted sum of the scores corresponding to the prior and generated attention before applying softmax.

$$\mathrm{Attention}(Q_f, K_f, V_f) = \mathrm{softmax}\left(\frac{Q_p K_p^\top}{\sqrt{D_{k_p}}}\right) V_p \oplus \mathrm{softmax}\left(\frac{Q_g K_g^\top}{\sqrt{D_{k_g}}}\right) V_g \tag{2}$$

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.

3 Method

3.1 Valid Division of Sentence

We input Chinese text x and use the word splitting tool to search for all possibilities of word splitting. In the sentence separation, some single characters and multi-character combinations are not words. If our model encounters the case of not words, it adds one score to the result of that separation. We select the lowest scores as the final results. If we get l divisions of the sentence, each division has k^l words. Each word in each division of is w_k^l.

$$w_k^l = \{[w_1^1;w_2^1;...;w_{K_1}^1][w_1^2;w_2^2;...;w_{K_2}^2]..[w_1^l;w_2^l;...;w_{K_l}^l]\} \tag{3}$$

Figure 1 shows three valid divisions of an example sentence.

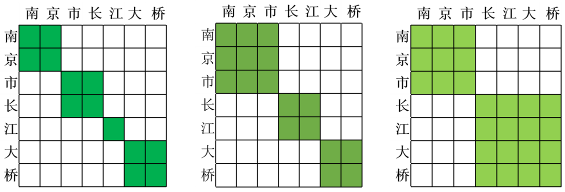

Figure 1: Word block attention.

In Figure 1, the Chinese text "南京市长江大桥(Nanjing Yangtze River Bridge)" is divided by the jieba and other word breaker to search for the possibility of all particours: [‘南京(Nanjing)’，‘南京市(Nanjing City)’，‘京市(Jing City)’，‘市长(Major)’，‘长江(Yangtze River)’，‘长江大桥(Yangtze River Bridge)’，‘江(River)’，‘大桥(Big Bridge)’], and then count each parting result, choosing the lowest score of the parting results: [南京(Nanjing)|市

长(Major)|江(River)|大 桥(Big Bridge)， 南京 市(Nanjing City)|长 江(Yangtze River)|大桥(Big Bridge)，南京市(Nanjing City)|长江大桥(Yangtze River Bridge)].

3.2 Attention of Block

A sentence can be viewed as a sequence of words，so each character embedding with its word in one division can be encoded in block local attention. Each character assigns to self-attention by characters in its local block instead of characters in the whole sentence, as shown in Figure 1.

Block embedding. We use BERT for character and word vector training. In each block, the character vector $(x_i)^l_k$ is stitched together with its word vector x^l_k respectively. Then we get new character vector $(\hat{x}_i)^l_k$ with word-lever information.

$$(\hat{x}_i)^l_k = (x_i)^l_k \oplus x^l_k \tag{4}$$

Attention compute. A multi-divide word block self-attention calculation results in attention matrix $(A_{i,j})^l_k$.

$$(A_{i,j})^l_k = \left(\frac{(Q_i)^l_k((K_j)^l_k)\top}{\sqrt{D_(K_j)^l_k}} \right) \tag{5}$$

Where $(Q_i)^l_k, (K_i)^l_k$ is calculated by query value, key value of the vector $(\hat{x}_i)^l_k$ for block attention; $D_{(K_j)^l_k}$ is the dimension of $(K_j)^l_k$.

The attention of each character in each block is $(b_i)^l_k$.

$$(b_i)^l_k = \text{softmax}((A_{i,j})^l_k)(V_i)^l_k \tag{6}$$

Where $(V_i)^l_k$ is calculated by extraction value of the vector $(\hat{x}_i)^l_k$ for block attention.

We use BERT for character training to get vector x_i for global attention. The attention of each character in global is g_i.

$$g_i = \text{softmax}(A_{i,j})V_i \tag{7}$$

Where $A_{i,j}$ is calculated by attention matrix of the vector x_i for global attention; V_i is calculated by extraction value of the vector x_i for global attention.

We compute attention with prior in Y. The words in each partition are individually calculated to form a blocked local attention mechanism in conjunction with the global attention mechanism by Equation (8).

$$Y = g \oplus b^1 \oplus b^2 \oplus ...b^L \tag{8}$$

Where $b_i{}^l = (b_i)^l_1 \cup (b_i)^l_2 \cup ...(b_i)^l_{K_l}$, for attention of each block in each division of sentence; $b^l = b_1{}^l \cup b_2{}^l \cup ...b^l_{K_l}$, for attention of each division; $g = (g_1, g_2, ..., g_n)$, for global attention.

In the model, we compute attention with prior in $\hat{Y}$. The incorporation $\oplus$ in Equation (8) is shown in details in Equation (9).

$$\hat{Y} = \sum_{l=0}^{L} p^l * b^l \tag{9}$$

Where $p^l \in [0,1]$ is a calculated probability , which balances the probability of global attention and each local attention; $\sum_{i=0}^{L} p^l = 1$; $b^0 = g$.

3.3 Design the CWA Model

We design the attention mechanism model to determine the extent of each block of attention. Multiple sequences of characters containing tokens are pretrained to obtain separate sets of character vectors. The global attention calculation is performed on the character vector to obtain the global weights and the word block attention calculation to obtain the local weights. The attention weights are computed separately for each character in each word to form a block local attention mechanism and combined with the global attention mechanism to input the model to obtain the results. Figure 2 is an example for details.

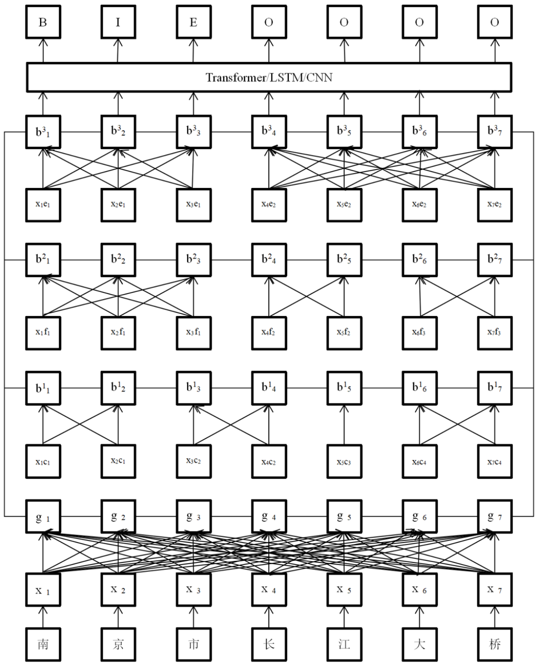
B
I
E
O
O
O
O
Transformer/LSTM/CNN
b^3_1 b^3_2 b^3_3 b^3_4 b^3_5 b^3_6 b^3_7
x_1e_1 x_2e_1 x_3e_1 x_4e_2 x_5e_2 x_6e_2 x_7e_2
b^2_1 b^2_2 b^2_3 b^2_4 b^2_5 b^2_6 b^2_7
x_1f_1 x_2f_1 x_3f_1 x_4f_2 x_5f_2 x_6f_3 x_7f_3
b^1_1 b^1_2 b^1_3 b^1_4 b^1_5 b^1_6 b^1_7
x_1c_1 x_2c_1 x_3c_2 x_4c_2 x_5c_3 x_6c_4 x_7c_4
g_1 g_2 g_3 g_4 g_5 g_6 g_7
x_1 x_2 x_3 x_4 x_5 x_6 x_7
南
京
市
长
江
大
桥

Figure 2: Chinese word attention model.

In Figure 2, the character sequences of [‘南(South)’, ‘京(Capital)’, ‘市(City)’, ‘长(Long)’,

‘江(River)’ , ‘大(Big)’, ‘桥(Bridge)’], [‘南(South)’, ‘京(Capital)’, ‘[s]’, ‘市(City)’, ‘长(Long)’, ‘[s]’, ‘江(River)’, ‘[s]’, ‘大(Big)’, ‘桥(Bridge)’], [‘南(South)’, ‘京(Capital)’, ‘市(City)’, ‘[s]’, ‘长(Long)’, ‘江(River)’, ‘[s]’, ‘大(Big)’, ‘桥(Bridge)’], [‘南(South)’, ‘京(Capital)’, ‘市(City)’, ‘[s]’, ‘长(Long)’, ‘江(River)’, ‘大(Big)’, ‘桥(Bridge)’], using BERT for pre-training, result in character vector groups $(x_1, x_2, x_3, x_4, x_5, x_6, x_7)$ respectively.

Boundary markers [s] are embedded between the words and input to the attention mechanism model to determine the range of attention for each block. We get {南京(Nanjing)[s]市长(Major)[s]江(River)[s]大桥(Big Bridge)，南京市(Nanjing City)[s]长江(Yangtze River)[s]大桥(Big Bridge)，南京市(Nanjing City)[s]长江大桥(Yangtze River Bridge)}. We input it as the multi-divide sequence of words [’南京(Nanjing)’, ’[s]’, ’市长(Major)’, ’[s]’, ’江(River)’, ’[s]’, ’大桥(Big Bridge)’]， [’南京市(Nanjing City)’, ’[s]’, ’长江(Yangtze River)’, ’[s]’, ’大桥(Big Bridge)’]， [’南京市(Nanjing City)’, ’[s]’, ’长江大桥(Yangtze River Bridge)’], using BERT for pre-training, result in word vector groups (c_1, c_2, c_3, c_4), (f_1, f_2, f_3), (e_1, e_2).

Each character vector is stitched together with its word vector by Equation (4), resulting in the incorporating vector $(x_1c_1, x_2c_1, x_3c_2, x_4c_2, x_5c_3, x_6c_4, x_7c_4)$, $(x_1f_1, x_2f_1, x_3f_1, x_4f_2, x_5f_2, x_6f_3, x_7f_3)$, $(x_1e_1, x_2e_1, x_3e_1, x_4e_2, x_5e_2, x_6e_2, x_7e_2)$.

Global attention calculation of character vector groups results in a weight of $(g_1, g_2, g_3, g_4, g_5, g_6, g_7)$, and a multi-divide word block attention calculation results in a weight of $(b_1^1, b_2^1, b_3^1, b_4^1, b_5^1, b_6^1 b_7^1)$, $(b_1^2, b_2^2, b_3^2, b_4^2, b_5^2, b_6^2, b_7^2)$, $(b_1^3, b_2^3, b_3^3, b_4^3, b_5^3, b_6^3, b_7^3)$.

The results are obtained by entering the Transform/LSTM/CNN model with labels.

4 Experiment

4.1 Setup

Datasets. The CWA 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).

Tag	P_{BRET}	P_{CWA}	R_{BRET}	R_{CWA}	$F1_{BRET}$	$F1_{CWA}$	Support
LOC	95.65	95.19	94.30	94.19	94.97	94.68	2752
ORG	86.57	87.98	91.81	92.04	89.11	89.97	1257
PER	93.93	95.54	96.29	95.26	95.10	95.40	1349
avg / total	93.09	93.59	94.21	93.95	93.59	93.76	5358

Table 1: Our performance on MSRA comparing with BERT-based.

Evaluation. We use P, R and F1 to evaluate our performance on MSRA, OntoNotes and Resume datasets comparing with BERT-base and other methods. We used F1 to evaluate our performance on the NE, NM and Overall of Weibo dataset comparing with BERT-base and other methods.

Model settings. For CWA model, we adopted similar settings as BERT-NER (Devlin et al., 2019). We download the specified pretrained BERT model provided by huggingface. We use Chinese-BERT-Base for Chinese task.

4.2 Compatibility with BERT

We compare CWA with BERT on MSRA datasets. Results are shown in Table 1.

Table 1 indicates that the MSRA data suppport 5358 tags incluing 2752 LOC, 1257 ORG and 1349 PER. We find that, for tags like ORG and PER, CWA+BERT can have a improvement over BERT on F1. But for LOC, it is opposite. Above all, the avg/total of CWA+BERT can have a improvement over BERT on F1 and P. But for R, it is opposite.

We also conducted experiments on the four datasets to further verify the effectiveness of CWA in combination with pre-trained model. The results are shown in Tables 2−5. In these experiments, we first use BERT encoders to obtain the word representations of each sequence, and then concatenate them into the character representations.

4.3 Effectiveness Study

Tables 2−5[1] show results on the MSRA, OntoNotes, Resume and Weibo datasets respectively against the compared baselines.

In Tables 2−5, compared methods include the best statistical models on these data set, which leveraged rich handcrafted features (Chen et al.,

[1] In Table 2−5, $*$ indicates that the model uses external labeled data for semi-supervised learning. † means that the model also uses discrete features.

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
Li et al. (2020)* †	92.46	**93.77**	**93.11**
BERT-base	93.09	**94.21**	93.59
CWA+BERT	**93.59**	93.95	**93.76**

Table 2: 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
Li et al. (2020)* †	74.73	76.70	**75.70**
BERT-base	76.01	**79.96**	77.93
CWA+BERT	**76.50**	79.95	**78.19**

Table 3: Performance on OntoNotes.

2006; Zhang et al., 2006; Zhou et al., 2013), character embedding features (Lu et al., 2016; Peng and Dredze, 2016a), 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). From the tables, we can see that the performance of the CWA method is significant better than other baseline methods on all four datasets. Compring with BERT, we find that, for MSRA, OntoNotes and Resume datasets, CWA+BERT can have a improvement over BERT on F1 and P. But for R, it is opposite. For Weibo dateset, CWA+BERT can have a improvement over BERT on NE and Overall. But for NM, it is opposite.

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
Li et al. (2020)* †	95.71	**95.77**	**95.74**
BERT-base	94.87	**96.50**	95.68
CWA+BERT	**96.50**	95.33	**95.91**

Table 4: Performance on Resume.

Models	NE	NM	Overall
Peng and Dredze, 2015	51.96	61.05	56.05
Peng and Dredze, 2016a*	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
Li et al. (2020)* †	**61.67**	**65.27**	**63.42**
BERT-base	57.58	**65.97**	62.07
CWA+ BERT	**65.77**	62.05	**63.80**

Table 5: Performance on Weibo. NE, NM and Overall denote F1 scores for named entities, nominal entities (excluding named entities) and both, respectively.

Models	MSRA	OntoNotes	Resume	Weibo
CWA	93.76	78.19	95.91	63.80
- Valid Division	93.69	78.06	95.75	63.44
- Word Attention	93.61	77.95	95.72	62.17

Table 6: An ablation study of the proposed model.

4.4 Ablation Study

To investigate the contribution of each component of our method, we conduct ablation experiments on all four datasets, as shown in table 6.

In the "- Valid Division" experiment, we remove the "Valid Division" group in CWA, as in word lexicons methods which incorporate many invalid words. The degradation in performance on all four datasets indicates the importance of the valid division of sentence, and confirms the advantage of our method.

In the "- Word Attention" experiment, we remove the "Word Attention" group in CWA, as in BERT without block local attention. The degradation in performance on all four datasets indicates the importance of the word attention, and confirms the advantage of our method.

5 Conclusion

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

References

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*.

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.

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*.

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.

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).

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.

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.

Xinnian Mao, Yuan Dong, Saike He, Sencheng Bao, and Haila Wang. 2008. Chinese word segmentation and named entity recognition based on conditional random fields. In *Proceedings of the Sixth SIGHAN Workshop on Chinese Language Processing*.

Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, and Alexander Ku. 2018. Image transformer. *CoRR*, abs/1802.05751.

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. 2016a. Improving named entity recognition for chinese social media with word segmentation representation learning. In *ACL*, page 149.

Nanyun Peng and Mark Dredze. 2016b. Learning word segmentation representations to improve named entity recognition for chinese social media. *CoRR*, abs/1603.00786.

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.

Yi Tay, Dara Bahri, Liu Yang, Donald Metzler, and Da-Cheng Juan. 2020. Sparse sinkhorn attention. *CoRR*, abs/2002.11296.

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*.

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

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*.

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.

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.