Attention Adaptive Chinese Named Entity Recognition Based on Vocabulary Enhancement

Ping Zhao, Quansheng Dou, Ping Jiang

Attention Adaptive Chinese Named Entity Recognition Based on Vocabulary Enhancement

Ping Zhao[1], Quansheng Dou[2] and Ping Jiang[2]

[1] School of Information and Electronic Engineering, Shandong Technology and Business University, 264000, Yantai, China
[2] School of Computer Science and Technology, Shandong Technology and Business University, 264000, Yantai, China
li_dou@163.com

Abstract. To deal with the lack of word information in character vector embedding and the problem of Out-of-Vocabulary in Named Entity Recognition, an attention adaptive chinese named entity recognition(CNER) model based on vocabulary enhancement(ACVE) is proposed. The mechanism of potential information embedding is designed, which acquires word-level potential information by constructing semantic vectors, and the fusion embedding of character information and word-level information realizes the enhancement of semantic features; We also propose an attention mechanism for adaptive distribution, which adaptively adjusts the position of attention by introducing a dynamic scaling factor to obtain the attention distribution suitable for NER tasks. Experiments on a special field dataset with a large number of out-of-vocabulary(OOV) words show that, compared with state-of-the-art methods, our method is more effective and achieves better results.

Keywords: Chinese Named Entity Recognition, Attention Mechanism, Adaptive Distribution, Scaling Factor.

1 Introduction

Named entity recognition(NER) is used to identify the entities with specific meanings in natural languages, such as names of people, places, organizations and proper nouns, etc. NER is related to the oriented natural language. Compared with English, Chinese characters, words, and grammatical structures are more complex, so it is more difficult to identify named entities. When there are OOV words in the text, the model is not sufficiently learned, which often leads to entity recognition errors due to ambiguity. What's more, in Chinese sentences, there are no natural separators between words, and the boundaries are not clear. CNER based on character-level embedding can avoid word segmentation errors, but this method cannot make full use of word information and the recognition effect is not satisfactory.

In this paper, the ACVE model is proposed with the following contributions: Potential information embedding(PIE) mechanism is implemented, which uses the new word discovery strategy to extract entities from related texts and expand the diction-

ary, to reduce the influence of OOV words on the performance of the model. The semantic vector of related words is obtained by matching characters and dictionary, which increases the utilization rate of potential information and helps the model to capture deep features; An attention mechanism for adaptive distribution(ADM) is implemented. By introducing a dynamic scaling factor, the model obtains the attention distribution suitable for NER tasks. It can dynamically adjust the attention position and adaptively focuses on the entity part.

2 Related Work

For the CNER problem, the most common deep learning method is to segment the input sentence with the CWS system and then apply the word-level sequence labeling model [1]. This framework makes the task easy to perform, but if the word segmentation is wrong, the sequence labeling will also cause errors. Luo and Yang [2] used multiple word segmentation results as additional features of the NER model, but they did not realize the incorrect segmentation problem caused by the CWS system. Zhang and Yang [3] proposed the Lattice LSTM model, which integrated the matching vocabulary information into the character sequence. Although it showed good results, there was a problem of information loss in the process of fusion. To reduce this problem, some scholars [4-5] transformed the word information fusion process from chain structure to graph structure and encoded it with graph neural networks(GNN), and proposed cooperative graph network(CGN) and lexicon-based graph neural(LGN) model to enhance the ability of global information capture. Although CGN and LGN models can capture the sequence structure of NER, they usually required RNN as the underlying encoder to capture the sequence, and the model structure was more complex. Liu et al. [6] introduced several simple selection strategies to match words for the model from a pre-prepared dictionary. However, these strategies did not consider sentence context.

3 ACVE Model

3.1 Symbols and Definitions

Let $C=\{c_1, c_2, \cdots, c_m\}$, $W = \{w_1, w_2, \cdots, w_D\}$ represent Chinese characters and vocabulary sets respectively. $\forall w_i \in W$, $w_i = c_1 c_2 \cdots c_k$, $k \geq 1$, $c_j \in C$, $j = 1, \cdots, k$.

Suppose $S = w_1 w_2 \cdots w_N = c_1 c_2 \cdots c_T$ is a natural sentence, $w_i \in W$, $c_j \in C$, $N \leq T$. A segment of consecutive characters in S is called a substring of S, and all the substrings of S constitute a set:

$$\Omega_s = \left\{ c_i c_{i+1} \cdots c_{i+h} \mid i \geq 1, i+h \leq T \right\} \tag{1}$$

Suppose z is a continuous character string in the corpus and all the character sets adjacent to the left and right of z in the corpus are respectively denoted as N_{left}^{z} and

N_{right}^{z}, which are called the left and right-adjacent character set of z.

Definition 1. Suppose S is a natural sentence, Ω_s is the set of S substrings, for $\forall x, y \in \Omega_s$, let $x{=}c_k \cdots c_{k+h}$ and $y{=}c_j \cdots c_{j+l}$. If $k+h<j$ or $j+l<k$, then the substrings x and y are not intersected, i.e. $x \cap y = \varnothing$.

Definition 2. Based on definition 1, x, y are two substrings of S and $x \cap y = \varnothing$, $\mathrm{MI}(x, y)$ called the mutual information(MI) of x and y.

$$\mathrm{MI}(x, y) = \log_2 \frac{p(x, y)}{p(x)p(y)} \tag{2}$$

Where $p(x)$ and $p(y)$ represent the probability for x and y appearing in the corpus separately, and $p(x, y)$ is the probability for x and y simultaneously appearing in the corpus.

Definition 3. Let z be a continuous string in the corpus, N_{left}^{z} and N_{right}^{z} be the left and right adjacent character sets of the string z respectively. Left and right adjacency entropy of substring z is calculated in formula (3), where $p(c \mid z)$ represents the conditional probability that c is the left and right adjacent characters of the string z.

$$\mathrm{BE_L}(z) = -\sum_{c \in N_{left}^{W}} p(c \mid z) \log p(c \mid z) \quad \mathrm{BE_R}(z) = -\sum_{c \in N_{right}^{W}} p(c \mid z) \log p(c \mid z) \tag{3}$$

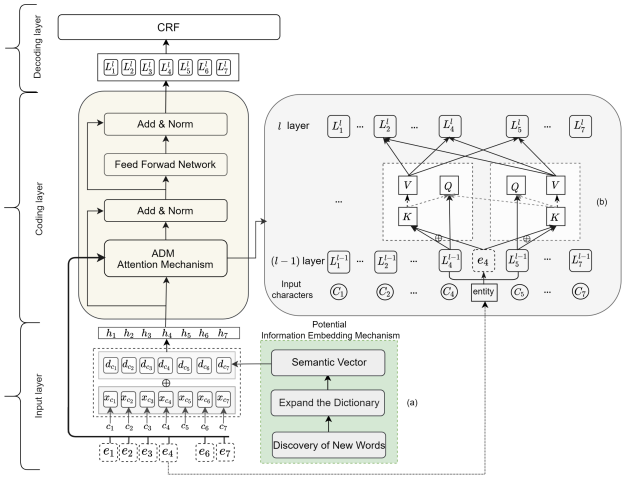

Fig. 1. ACVE model architecture mainly includes two parts: (a) Potential information embedding mechanism. (b) Attention mechanism of adaptive distribution.

3.2 Overall Architecture

The ACVE model is divided into three layers, including the input layer, encoding layer, and decoding layer, as shown in Figure 1. Given statement

$S = c_1 c_2 \cdots c_T, c_i \in C, i = 1, \cdots, T$. The input layer obtains the word embedding vector $\boldsymbol{X} = \{\boldsymbol{x}_{c_1}, \cdots, \boldsymbol{x}_{c_T}\}$ corresponding to the sentence S through BERT [7]. For $\forall c_i \in S$, the corresponding semantic vector $\boldsymbol{d}_{c_i}$ is generated by the PIE mechanism and $\boldsymbol{d}_{c_i}$=PIE(c_i) . The semantic vector $\boldsymbol{d}_{c_i}$ contains the word-level potential information related to the character. The word embedding vector $\boldsymbol{x}_{c_i}$ is merged with the semantic vector $\boldsymbol{d}_{c_i}$ to obtain the embedding vector $\boldsymbol{h}_i$. The formula is as follows:

$$\boldsymbol{h}_i = \boldsymbol{x}_{c_i} + \boldsymbol{W}_{c_i} \boldsymbol{d}_{c_i} \tag{4}$$

Matrix $\boldsymbol{H} \in \mathbb{R}^{T \times H_c}$ is denoted as $\boldsymbol{H} = \{\boldsymbol{h}_1, \cdots, \boldsymbol{h}_T\}$. The matrix $\boldsymbol{H}$ contains character features and word-level information. The next step will be to extract features of $\boldsymbol{H}$ through the coding layer. The coding layer uses a Transformer as the encoder, and an attention mechanism of adaptive distribution(ADM) is proposed based on this encoder. ADM attention mechanism introduces a scaling factor to adjust the probability distribution of the output. Matrix $\boldsymbol{H} = \{\boldsymbol{h}_1, \cdots, \boldsymbol{h}_T\}$ passes through the encoding layer to get the hidden output $\boldsymbol{L}^l = \{\boldsymbol{L}_1^l, \cdots, \boldsymbol{L}_T^l\}$. The matrix $\boldsymbol{L}^l$ contains the local and global features of the input characters. The hidden output $\boldsymbol{L}^l$ is sent to CRF. The calculation is shown in formula (5). The above formula $\boldsymbol{w}_k \in \mathbb{R}^{H_c}$ and b_k are trainable parameters specific to the k-th label and r is the number of different NER labels.

$$p(k \mid c_i) = \frac{\exp(\boldsymbol{w}_k^{\top} \boldsymbol{L}_i^l + b_k)}{\sum_{j \in \{1, \ldots, r\}} \exp(\boldsymbol{w}_j^{\top} \boldsymbol{L}_i^l + b_j)} \tag{5}$$

The PIE and ADM mechanisms involved in the model will be introduced below.

3.3 Potential Information Embedding Mechanism

PIE mechanism needs the help of dictionary, so we use new word discovery based on mutual information(MI) and adjacency entropy(BE) to expand the dictionary. The input of the algorithm is a preprocessed corpus M, M=$c_1, \cdots, c_N$, $c_j \in C$. The MI and BE threshold are set to be MI_{th} , BE_{th} . The algorithm is described as follows.

If the current character is c_i , then $\mathrm{MI}(c_i, c_{i+1})$ is calculated. If $\mathrm{MI}(c_i, c_{i+1})$ is greater than MI_{th} , connect the current character to the right adjacent character to form a string $c_i c_{i+1}$, and calculate the $\mathrm{MI}(c_i c_{i+1}, c_{i+2})$ value of the string $c_i c_{i+1}$ and the right adjacent character c_{i+2} until the MI between the string $c_i c_{i+1} \cdots c_k$ and c_{k+1} is less than the MI_{th} , and stop extending to the right and mark $c_i c_{i+1} \cdots c_k$ as a candidate. Filter candidate words by BE. If the current candidate word is $c_i c_{i+1} \cdots c_k$, then $\mathrm{BE}_{\mathrm{L}}(c_i c_{i+1} \cdots c_k)$ and $\mathrm{BE}_R(c_i c_{i+1} \cdots c_k)$ of the $c_i c_{i+1} \cdots c_k$ are calculated. If both are greater than BE_{th} , the candidate word is retained, otherwise it is deleted. A new word set *CanList* is obtained

by the algorithm. We remove the common words from the jieba8 dictionary and merge the remaining words with the *CanList* to obtain dictionary ε_{ent}. If $\exists wd \in \varepsilon_{ent}$ and c_i is at the beginning, middle or end of wd, set the first, second, or third component of semantic vector $\boldsymbol{d}_{c_i}$ to be 1, otherwise 0. If there is no word containing c_i in ε_{ent}, set the fourth component of $\boldsymbol{d}_{c_i}$ to be 1, otherwise 0. The purpose of the PIE mechanism is to obtain a semantic vector $\boldsymbol{d}_{c_i}$. It contains word-level potential information related to the character.

3.4 Attention Mechanism of Adaptive Distribution

For scaled dot-product attention, Yan [8] had proved that this attention had a poor effect on NER. Therefore, we designed an ADM attention mechanism, which adjusted the probability distribution by introducing a dynamic scaling factor to prevent the inner product from being too large and adaptively weighted the position of the attention.

We used the same attention transformation as in literature [9] to achieve entity enhancement. Given the hidden representation of a sequence $\left\{\boldsymbol{L}_1^{l-1},\cdots,\boldsymbol{L}_T^{l-1}\right\}$ for the $(l-1)$ th layer and packed together as the matrix $\{\boldsymbol{L}_t^{l-1}\}_{t=1}^{T} \in \mathbb{R}^{T\times H_c}$. The query matrix $\boldsymbol{Q}^l = \left\{\boldsymbol{q}_t^l\right\}_{t=l}^{T}$ is obtained by multiplying the matrix $\{\boldsymbol{L}_t^{l-1}\}_{t=1}^{T}$ by $\boldsymbol{W}_{L,q}^l$. The key matrix $\boldsymbol{K}^l = \left\{\boldsymbol{k}_t^l\right\}_{t=1}^{T}$ and value matrix $\boldsymbol{V}^l = \left\{\boldsymbol{v}_t^l\right\}_{t=1}^{T}$ of the l th layer of the attention mechanism is calculated as follows formula(6), where $\boldsymbol{W}_{L,q}^l$, $\boldsymbol{W}_{L,k}^l$, $\boldsymbol{W}_{L,v}^l \in \mathbb{R}^{T\times H_c}$ is the trainable parameter of l th layer and $\boldsymbol{W}_{e,k}^l$, $\boldsymbol{W}_{e,v}^l \in \mathbb{R}^{H_e\times H_c}$ is the trainable parameter of related entities. $\boldsymbol{E}_{ent}$ is an entity embedded query table.

The probability distribution of Softmax output is adjusted by the dynamically learnable scaling factor η. A set of entity embedding $\left\{\boldsymbol{E}_{ent}[e_1], \cdots, \boldsymbol{E}_{ent}[e_t]\right\}$ is represented by $\boldsymbol{e} \in \mathbb{R}^{T\times H_e}$ integration, and the attention score of the i-th character in the l th layer is calculated as shown in formula(7), where $\boldsymbol{w}_1 \in \mathbb{R}^{H_c}$, $\delta \in \mathbb{R}$ is a trainable parameter.

$$\boldsymbol{k}_t^l = \begin{cases} \boldsymbol{L}_t^{l-1^{\mathrm{T}}}\boldsymbol{W}_{L,k}^l \ \text{ if } e_t = 0, \\ \frac{1}{2}(\boldsymbol{L}_t^{l-1^{\mathrm{T}}}\boldsymbol{W}_{L,k}^l + \boldsymbol{E}_{ent}^{\mathrm{T}}[e_t]\,\boldsymbol{W}_{e,k}^l) \ \text{ else;} \end{cases} \quad \boldsymbol{v}_t^l = \begin{cases} \boldsymbol{L}_t^{l-1^{\mathrm{T}}}\boldsymbol{W}_{L,v}^l \ \text{ if } e_t = 0, \\ \frac{1}{2}(\boldsymbol{L}_t^{l-1^{\mathrm{T}}}\boldsymbol{W}_{L,v}^l + \boldsymbol{E}_{ent}^{\mathrm{T}}[e_t]\,\boldsymbol{W}_{e,v}^l) \ \text{else;} \end{cases} \tag{6}$$

$$\boldsymbol{S}_i^l = \mathrm{softmax}\left\{\frac{\boldsymbol{q}_i^l \boldsymbol{K}^{l^{\mathrm{T}}}}{\sqrt{\eta}}\right\} = \mathrm{softmax}\left\{\frac{\boldsymbol{q}_i^l(\boldsymbol{L}^{l-1}\boldsymbol{W}_{L,k}^l + \mathrm{e}\boldsymbol{W}_{e,k}^l)^{\mathrm{T}}}{2\sqrt{\eta}}\right\} \tag{7}$$

$$\eta = \min(\mathrm{ReLu}(\boldsymbol{w}_1 \boldsymbol{L}^{l-1^{\mathrm{T}}} + \delta), \sqrt{H_c/n}) + 1; \tag{8}$$

In formula (8), the dynamic scaling factor η linearly activates the matrix $\boldsymbol{L}^{l-1}$ containing feature information through the ReLU function. The output value is in the range of $[0,\infty)$ and η is bounded in the range of $\left[1,1+\sqrt{H_c/n}\right]$. By using the sparse activation of the ReLU function, the scaling factor can be adjusted without increasing the computational cost.

4 Experiments

4.1 Experimental Setup

This article conducts experiments on 4 datasets, including Novel, Medicine (CCKS2018), Weibo, and Resume dataset. "The Legend of the Condor Heroes" is selected as the novel corpus. Weibo dataset contains four types of entities, all entities are also divided into named entities(NE) and generic entities(NM). The hyperparameter settings are as follows: coding layers l=12 , self-attention $A=12$, character hiding size $H_c=768$, entity hiding size $H_e=64$, initial learning rate $3e^{-5}$, epoch number 3, max sentence length 200, and batch size 32.

4.2 Ablation Study

To verify the effectiveness of PIE and ADM, we use the LSTM + CRF as the baseline model and carried out experiments on Weibo and Novel datasets. By adding PIE and ADM mechanisms to the baseline model, the effectiveness of the two structures is proved step by step. The experimental results of the control group are shown in Table 1.

On the Weibo dataset, after Step1"+PIE" ("+" means adding a model), compared with the baseline, the F1 value increased by 7.5%, and the recall rate increased by 3.84%. When Step2 "+ADM" is completed, compared with the baseline, the recall rate increased by 9.14%, and the F1 value increased by 8.2%. On the Novel dataset, after Step1"+PIE", the F1 value increased by 6.68%, and the recall rate increased by 1.9%. When Step2 "+ADM" is completed, compared with the baseline, the F1 value increases by 8.09%, the recall rate increases by 4.24%. From the perspective of decomposition, Step1 "+PIE" has a greater effect on improving the F1 value, and step2 "+ADM" is very important for improving the recall rate. Therefore, the two parts of PIE and ADM are complementary.

Table 1. Ablation experiment results of WeiboNER dataset and MSRA dataset.

Models	Weibo			Novel		
	P	R	F1	P	R	F1
Character embedding(baseline)	69.45	58.47	61.2 ± 0.42	66.45	60.23	59.10 ± 0.21
+Potential Information Embedding	70.10	62.31	68.7 ± 0.34	70.35	62.13	65.78 ± 0.04
+Adaptive Distribution Selection	72.07	67.61	69.4 ± 0.21	68.14	64.47	67.19 ± 0.24
FWPI+BERT	72.62	70.13	71.5 ± 0.20	77.75	73.67	75.26 ± 0.12

Table 2. Performance comparison table of different models.

Models	Medicine	Novel	Resume	NE	NM	Overall
Char-based(LSTM)[10]	84.71	59.11	92.41	50.25	55.29	52.77
BiLSTM+CRF[11]	85.09	62.74	93.26	53.95	62.63	58.96
Lattice LSTM[12]	87.90	63.89	94.46	58.04	61.25	59.79
FWPI	92.21	72.43	95.10	71.21	69.45	69.33

Table 3. Experimental results of F1 value of various attention mechanisms.

Models	Attention	Novel	Weibo	Models	Attention	Novel	Weibo
	Soft[11]	62.70	59.15		Soft[11]	70.05	69.72
	Self+Scaled[12]	63.93	60.12		Self+Scaled[12]	70.20	69.98
NER-only	Self+Uscaled[6]	64.73	61.54	Joint	Self+Uscaled[6]	71.09	70.14
	ADM	65.98	62.48		ADM	72.43	71.33

4.3 Comparison with Existing Methods

We conducted experiments on the four datasets mentioned above and compared the experimental results with the current common models. The experimental results are listed in Table 2. Although the BiLSTM+CRF model shows good results on the Resume dataset, it has not made significant improvements on the Novel and Weibo corpus. In addition to word segmentation errors on social media, another important reason is that the semantic expression of colloquial texts limits their performance. In contrast, our model has better advantages and performance in adapting to spoken texts. The ACVE model has a significant increase in the F1 value on the Novel and Weibo datasets.

4.4 Verification of Effectiveness of ADM Attention Mechanism

To verify the effectiveness of the ADM, Weibo and Novel datasets are selected for experiments, and BiLSTM+CRF(NER-only) and FWPI-ADM(Joint) are used as baseline models; Based on the NER-only and Joint models, various attention mechanisms are added to the experiment. The experimental results are shown in Table 3. Experimental results show that the Joint model is always better than the NER-only model. In the NER-only model, the recognition effect of ADM is better than that of Self+Scaled and Self+Unscaled. For all experiments, Self+Unscaled attention produced a better F1 value than Self+Scaled attention, which indicated that clear attention distribution was helpful to the NER task. The ADM attention mechanism adds a dynamic scaling factor based on Self+Unscaled attention, and its value is also increased by about 1% on the basis of Self+Unscaled attention, which proves the effectiveness of the ADM.

5 Conclusions

This paper proposed an ACVE model integrated with potential information. The PIE mechanism provides word-level deep features, and the semantic vector and character embedding vector are merged as the input layer to realize the enhancement of infor-

mation features; The ADM mechanism obtains an attention distribution suitable for CNER and extracts entity features without increasing calculation cost.

6 References

1. He, H., & Sun, X.: A unified model for cross-domain and semi-supervised named entity recognition in chinese social media. In Proceedings of the AAAI Conference on Artificial Intelligence.Vol. 31, No. 1. (2017).
2. Luo, W., & Yang, F.: An Empirical Study of Automatic Chinese Word Segmentation for Spoken Language Understanding and Named Entity Recognition. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies pp. 238–248. (2016).
3. Zhang, Y., & Yang, J.: Chinese NER Using Lattice LSTM. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) Vol. 1, pp. 1554–1564. (2018).
4. Sui, D., Chen, Y., Liu, K., Zhao, J., & Liu, S.: 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. pp. 3828–3838. (2019).
5. Gui T., Zou Y., Zhang Q.: A lexicon-based graph neural network for chinese ner. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 1040-1050. (2019).
6. Wei Liu., Tongue Xu., Qinghua Xu., Jiayu Song., Yueran Zu.: 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, pp. 2379–2389. (2019).
7. Devlin., Jacob, et al.: BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. 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), 2018, pp. 4171–4186. (2019).
8. Yan, H., Deng, B., Li, X., & Qiu, X.: TENER: Adapting Transformer Encoder for Named Entity Recognition. ArXiv Preprint ArXiv:1911.04474. (2019).
9. Jia, C., Shi, Y., Yang, Q., & Zhang, Y.: Entity enhanced BERT pre-training for Chinese NER. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP, pp. 6384-6396. (2020, November).
10. Lample, Guillaume, et al.: Neural Architectures for Named Entity Recognition. 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016-Proceedings of the Conference, pp. 260–270. (2016).
11. Z.Huang., W.Xu., K.Yu.: Bidirectional LSTM-CRF Models for Sequence Tagging. arXiv preprint arXiv:1508.01991, (2015).
12. Zhang, Y., Yang, J.: Chinese NER Using Lattice LSTM. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) Vol. 1, pp. 1554–1564. (2018).