Is Local Window Essential for Neural Network based Chinese Word Segmentation ?

Jinchao Zhang Fandong Meng Mingxuan Wang Daqi Zheng
Wenbin Jiang Qun Liu

Key Laboratory of Intelligent Information Processing,
Institute of Computing Technology, Chinese Academy of Sciences
{zhangjinchao, mengfandong, wangmingxuan, zhengdaqi,
jiangwenbin, liuqun}@ict.ac.cn

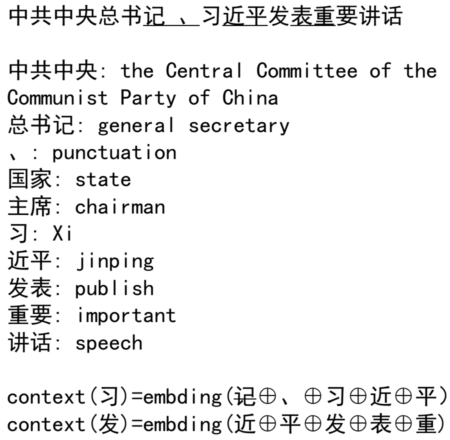
Abstract. Neural network based Chinese Word Segmentation (CWS) approaches can bypass the burdensome feature engineering comparing with the conventional ones. All previous neural network based approaches rely on a local window in character sequence labelling process. It can hardly exploit the outer context and may preserve indifferent inner context. Moreover, the size of local window is a toilsome manual-tuned hyper-parameter that has significant influence on model performance. We are wondering if the local window can be discarded in neural network based CWS. In this paper, we present a window-free Bi-directional Long Short-term Memory (Bi-LSTM) neural network based Chinese word segmentation model. The model takes the whole sentence under consideration to generate reasonable word sequence. The experiments show that the Bi-LSTM can learn sufficient context for CWS without the local window.

Keywords: Chinese Word Segmentation, Neural Network, Window

1 Introduction

Chinese word segmentation is to generate reasonable word sequence from non-delimited sentences. The most popular model is character labelling model [11, 9] with statistical supervised approach [1, 6], which assign positional labels (B, M, E, S) to characters according to the context. In these approaches, character context is represented by features that strongly depend on the handcrafted feature template. Although feature template can easily incorporate the linguistic knowledge, it is indeed a heavy burden to design an appropriate feature template due to the feature diversity and uncertain local window size.

In recent years, neural network models are introduced into CWS due to their ability to bypass the feature engineering. Zheng et al. [14] applied the architecture of SENNA [4] to CWS and pos-tagging to avoid the feature engineering and also speed up the training process with a perceptron-style algorithm. Pei et al. [8] proposed a Max-Margin Tensor Neural Networks for CWS and modeled the interactions of tag-tag, tag-character, character-charcter. Chen et al.



中共中央总书记 、习近平发表重要讲话

中共中央: the Central Committee of the
Communist Party of China
总书记: general secretary
、: punctuation
国家: state
主席: chairman
习: Xi
近平: jinping
发表: publish
重要: important
讲话: speech

context(习)=embding(记⊕、⊕习⊕近⊕平)
context(发)=embding(近⊕平⊕发⊕表⊕重)

Fig. 1. Restriction of local window.

[3] proposed a gated recursive neural network to model the complicated combinations of the contextual characters to simulate the feature template. Chen et al. [2] introduced the LSTM neural network for CWS to capture the potential long-distance dependency.

These previous neural network based models all rely on a local window. The local window provides the rigid context for a character, and makes it difficult to exploit the outer context which is proved useful by Chen et al. [2] and it may preserve useless information in context. For instance, in Figure 1, when labelling character "习", the character "记" in local window is useless due to the appearance of a caesura sign. Actually, no matter how long the window size is, the characters before the caesura sign are meaningless for labelling "习". On the contrary, longer the window size is, the more noise will be introduced from left context. However, when labelling "发", the character "话", which is helpful while is out of the local window. In addition, the local window size is an important hyper-parameter that apparently affect the model performance. It is costly to determine the most suitable size, which may change with the language and the segmentation criterion.

Long Short-Term Memory (LSTM) Neural Network[5] is a category of recurrent neural network with memory cell and gates, which endow it the ability to determine forget or memorize the historical information. We are wondering how well the bi-directional LSTM (Bi-LSTM) neural network can capture related context in whole sentence without local window. Therefore, we present a window-free Bi-LSTM neural network for CWS. The experiments show that Bi-LSTM based neural network can capture the sufficient related context for character sequence labelling. The local window is not essential in Bi-LSTM based CWS system.

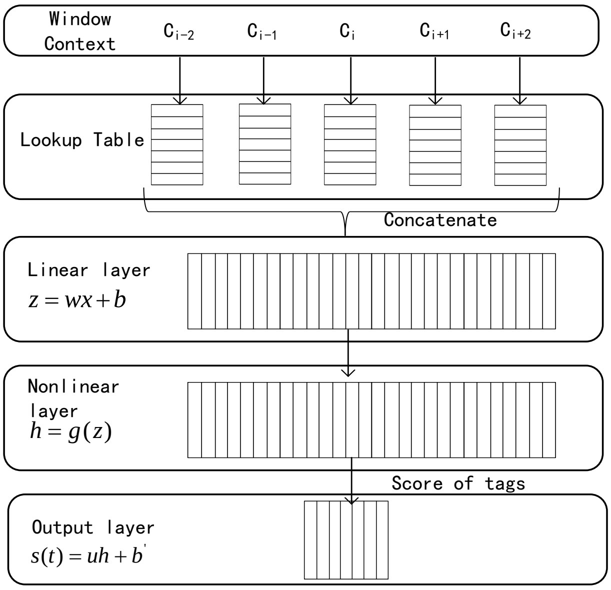


Fig. 2. Conventional Neural Model for Chinese Word Segmentation.

2 Local window based Neural Network for CWS

The character sequence labelling model is the most popular model for CWS. Traditional approaches are based on the feature template, while the neural network approaches are based on the character embedding. All of them rely on the local window to extract the context.

3 Conventional Neural Network for CWS

As Figure 2 shows, the conventional neural model for CWS consists of a lookup table layer, linear layers and nonlinear layers, and a output layer.

The lookup table transforms a character to a distributed real vector called embedding. The embeddings in local window are concatenated to form the context of a character. The linear layer combine the input vector and the nonlinear layer do the nonlinear transformation with *sigmoid* or *tanh* function and so on. The output layer computes the label probability distribution of each character. Finally, the best segmentation result is inferred by beam search algorithm or dynamic algorithm. In conventional neural network based models, the local context is the feature. As a result, local window is of great importance to the model performance.

3.1 LSTM Neural Network for CWS

LSTM was proposed to address the gradient vanish of recurrent neural network. A LSTM unit consists of a memory cell and three gates. c_t is the memory cell stores the content of content a unit. i_t is the input gate to control the scale of current input. f_t is the forget gate to control forget extent of the last content. o_t is the output gate to control the scale of the output. The LSTM unit can be formalised as:

$$i_t = \sigma(W_{xi} + W_{hi}h_{t-1}) \tag{1}$$

$$f_t = \sigma(W_{xf} + W_{hf}h_{t-1}) \tag{2}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot (W_{xc}x_t + W_{hc}h_{t-1}) \tag{3}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_t) \tag{4}$$

$$h_t = o_t \odot \tanh c_t \tag{5}$$

Chen et al. first introduced the LSTM unit into the Chinese word segmentation. However, their model also needs window to capture the right context.

4 Window-free Bi-LSTM Neural Network for CWS

Inspired by Chen et al. , we wonder whether the Bi-LSTM Neural network can thoroughly eliminate the local window or not. Therefore, we introduce the Bi-LSTM architecture into the CWS.

4.1 Model Architecture

Figure 3 shows our architecture. The first layer is the lookup table, which is identical to other previous works. For the absence of window, the character embeddings are transferred into the forward and backward layer without being concatenated. The forward layer accepts the positive sequence of embeddings, while the backward layer deals with the inverted sequence of embedding. These two layers work separately. Thus, we can get two hidden states $\overrightarrow{H}$ and $\overleftarrow{H}$, in which $\overrightarrow{H_i}$ and $\overleftarrow{H_i}$ represent the left context and the right context of character C_i. The two states are concatenated into one and transferred to the next layer that can be another Bi-LSTM layer or a logistic layer. The logistic layer transforms the input with a linear function, and then normalize the probabilities with a $softmax$ function.

4.2 Training

The parameter θ in our model contains: the lookup table, 6 parameter matrixes for each LSTM layer (equation 1-5), the parameter matrix of logistic layer. We

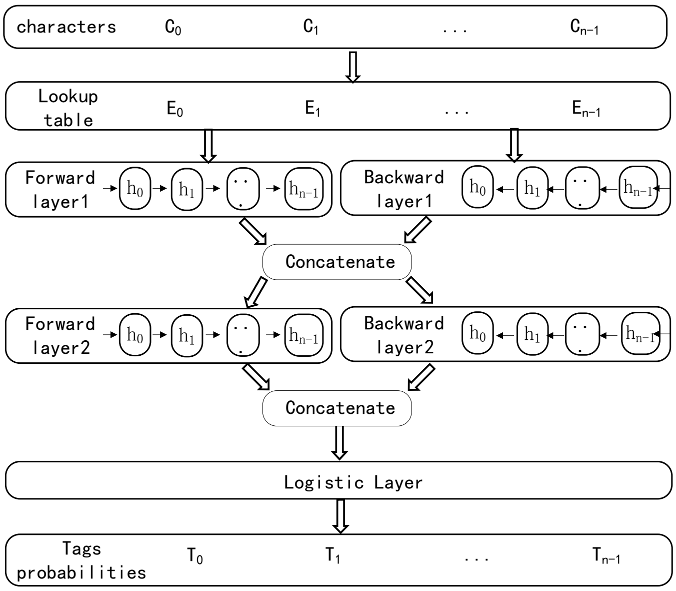


Fig. 3. Bi-directional LSTM model for Chinese Word Segmentation.

use the max-likelihood criterion to train our model, and we import the $L2$ regularization item into the objective function to prevent overfitting. The objective function is:

$$J(\theta) = \sum_{m=1}^{M} \sum_{n=1}^{N} log(P(tag|c)) + \frac{\lambda}{2}\|\theta\|^2 \tag{6}$$

where M is the sentence number, N is the sentence length, $P(tag|c)$ is the conditional probability, λ is the regularization constant.

To minimize $J(\theta)$, we use AdaDelta algorithm to update the parameters:

$$E[g^2]_t = \rho E[g^2]_{t-1} + (1-\rho)g_t^2 \tag{7}$$

$$\mathit{\Delta} x_t = -\frac{\sqrt{\mathit{\Sigma}_{\tau=1}^{t-1} \mathit{\Delta} x_\tau}}{\sqrt{E[g^2] + \epsilon}} g_t \tag{8}$$

where $E[g^2]_t$ is the average of the squared gradients at time t, and the ρ is the decay constant, ϵ is a constant added to better condition the denominator, g_t is the gradient of the parameters at time t. ρ and ϵ are hyper-parameter.

Table 1 shows all hyper-parameters. To make it fair, we follows setup in Chen et al.[2]

Character embedding size	$d = 100$
Bi-LSTM Layer number	$n = 2$
Hidden unit number	$h = 150$
L2-Regularization	$\lambda = 10^{-4}$
Dropout rate on input layer	$p = 0.2$
Decay constant in AdaDelta	$\rho = 0.95$
Constant in AdaDelta	$\epsilon = 10^{-6}$
Batch size	$b = 20$
Beam size	$beam = 20$

Table 1. Hyper-parameters of our model

Models	PKU	MSRA	CTB6
LSTM-No-Window	92.3	93.0	91.5
Bi-LSTM-Window	95.6	96.2	95.4
Bi-LSTM-No-Window	95.6	96.3	95.4
Chen et al. [2]	95.7	96.4	94.9

Table 2. Performance of our three models and the Chen et al. result, local window slightly affects the Bi-LSTM model, and enormously affect the LSTM model

Model	PKU	MSRA	CTB6
Mansur et al. [7]	93.0	-	-
Zheng et al. [14]	92.4	94.4	-
Pei et al. [8]	93.5	94.4	-
Chen et al. [3]	95.9	96.2	95.3
Chen et al. [2]	95.7	96.4	94.9
Bi-LSTM-No-Window	95.6	96.3	95.4

Table 3. Performance of window-free Bi-LSTM and previous works without extra knowledge. Our model achieves competitive result without local window

5 Experiments

5.1 Datasets

We run experiments on three widely used benchmark datasets, PKU, MSRA [10] and CTB6(LDC2007T36) [12]. For PKU and MSRA datasets, we take the first 90% sentences of the training data as training set and the rest 10% sentences as development set according to the previous works. For CTB6 dataset, we divide the training, development and test sets according to Yang et al. [13]. All datasets are preprocessed by replacing the Chinese idioms and the continuous English characters and digits with a unique flag. We adopt F1-score as the evaluation measure.

5.2 Experiment Results

We implement three models: LSTM-No-Window ,Bi-LSTM-No-Window, Bi-LSTM-Window. We investigate these three models on PKU, MSRA and CTB6 dataset.

We refer to the result of Chen et al. [2] as LSTM with window (LSTM-Windows). Table 2 shows the F1-Score of models. Compared with Bi-LSTM models and LSTM-Windows model, the LSTM-No-Window model achieves much lower F1-Score by a large gap. From that, we conclude that LSTM-No-Window is insufficient to capture context and the local window can provide abundant context information. However, on Bi-LSTM model, local window slightly affect the performance. On MSRA dataset, the window-free model performs better than window-based model. Therefore, we can conclude that the Bi-LSTM model has strong ability to capture related context information for Chinese word segmentation, which means local window is not essential for Bi-LSTM based model. Table 3 compares our result with the previous neural network based CWS models. Although our model does not have a local window to extract the context feature, it outperforms previous neural network model on CTB6 dataset, and ranks 2nd on MSRA dataset by a slightly lag(-0.1), and ranks 3rd by a little lag (-0.3, [3] is based on the recursive neural network). The results suggest that the window-free Bi-LSTM neural network can automatically capture the related context for Chinese word segmentation. Although the local window appears in every previous neural network based approaches, it is not essential for Bi-LSTM based one.

6 Related Work

The most popular model for CWS is the sequence labelling model proposed by Xue. [11]. The previous supervised methods [6, 1] rely on the handcrafted template feature. In recent years, neural network based approaches [7, 14, 8, 3] were proposed to avoid the feature engineering. All neural networks based CWS rely on the local window, which restricts the related context in a fixed scope. Our work is inspired by Chen et al. [2]. They exploit a LSTM neural network to capture the left context and a window to capture the right context. Our model exploits a Bi-LSTM neural network to capture the related context and discard the local window thoroughly and achieves comparable or even better performance.

7 Conclusion

In this paper, we present a window-free Bi-LSTM based Chinese word segmentation model. Compared with the local window based Bi-LSTM model, the window-free Bi-LSTM model achieves the identical or better performance. Compared with other previous local window based neural CWS approach, our model achieves the competitive or better performance. The result shows that Bi-LSTM neural network can automatically capture the related context and the local window is not essential for Bi-LSTM based CWS.

Acknowledgments

The research is supported by National Natural Science Foundation of China (Contract 61202216). Liu is partially supported by the Science Foundation Ireland (Grant 12/CE/I2267 and 13/RC/2106) as part of the ADAPT Centre at Dublin City University. We sincerely thank the anonymous reviewers for their thorough reviewing and valuable suggestions.

References

1. Berger, A., Pietra, S.D., Pietra, V.D.: A maximum entropy approach to natural language processing. Computational Linguistics 22(1), 39–71 (1996)
2. Chen, X., Qiu, X., Zhu, C., Liu, P., Huang, X.: Long Short-Term Memory Neural Networks for Chinese Word Segmentation. In: Proceedings of the Empirical Methods In Natural Language Processing (2015)
3. Chen, X., Qiu, X., Zhue, C., Huang, X.: Gated Recursive Neural Network for Chinese Word Segmentation. In: Proceedings of Annual Meeting of the Association for Computational Linguistics (2015)
4. Collobert, R., Weston, J., Léon, B., Michael, K., Koray, K., Pavel, K.: Natural Language Processing (almost) from Scratch. Journal of Machine Learning Research 1, 1–48 (2011)
5. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation 8(9), 1735–1780 (1997)
6. Lafferty, J., McCallum, A., Pereira, F.C.N.: Gated Recursive Neural Network for Chinese Word Segmentation. In: Proceedings of Annual Meeting of Eighteenth International Conference on Machine Learning (2015)
7. Mansur, M., Pei, W., Chang, B.: Feature-based Neural Language Model and Chinese Word Segmentation. In: Proceedings of International Joint Conference on Natural Language Processing (2013)
8. Pei, W., Ge, T., Chang, B.: Max-Margin Tensor Neural Network for Chinese Word Segmentation. In: Proceedings of Annual Meeting of the Association for Computational Linguistics (2014)
9. Peng, F., Feng, F., McCallum, A.: Chinese segmentation and new word detection using conditional random fields. In: Proceedings of International Conference on Computational Linguistics (2004)
10. T.Emerson: The second international Chinese word segmentation bakeoff. In: Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing. pp. 123–133 (2005)
11. Xue, N.: Chinese word segmentation as character tagging. Computational Linguistics and Chinese Language Processing 8(1), 29–48 (2003)
12. Xue, N., Xia, F., Chiou, F.D., Palmer, M.: The Penn Chinese TreeBank: Phrase structure annotation of a large corpus. Natural Language Engineering, 11(2), 207–238 (2005)
13. Yang, Y., Xue, N.: Chinese comma disambiguation for discourse analysis. In: Proceedings of Annual Meeting of the Association for Computational Linguistics (2012)
14. Zheng, X., Chen, H., Xu, T.: Deep Learning for Chinese Word Segmentation and POS Tagging. In: Proceedings of the Empirical Methods In Natural Language Processing (2013)