An Algorithm Based on Simple CNN and BI_LSTM Network for Chinese Word Segmentation

Xiaohan Guan, Xin Liu and Zhi Liu
North China University of Technology, Beijing, China

Abstract. In dealing with most Chinese NLP tasks, word segmentation is an indispensable and critical work, and it also affects the accuracy of subsequent tasks. Neither the traditional BI_LSTM network can extract features effectively, nor the CNN network can deal with the timing problem effectively. As a result, this paper proposes a new algorithm to solve these problems. First, the Chinese characters’ feature in a whole sentence is extracted and recombined using the CNN network; then, the recombined feature is combined in BI_LSTM network; finally, each word of the whole sentence which classified by weight sharing and the corresponding classification of each word is the consequent output. Not only the advantages of CNN network are utilized for feature extraction, but also the advantages of BI_LSTM network are retained for timing processing. In conclusion, the learning ability of BI_LSTM network is increased, and the accuracy of output is improved to 98%.

1. Introduction

In Chinese, each character is the smallest unit that exists independently and explains one single semantic. Phrases are more capable to expressing complicated semantics than single characters. Contrary to the English, which the space is used as separators; there is no obvious separator in Chinese. Therefore, the Chinese word segmentation is the most basic work in Chinese NLP tasks, and the quality of the segmentation affects the accuracy of the subsequent works.

The Chinese word segmentation is the premise in many Chinese NLP tasks including machine translation, text summary, information retrieval etc. From the perspective of machine learning, the word segmentation can be regarded as a sequence classification task, in which each element must be correspondingly classified. The methods commonly used in traditional word segmentation include Maximum Entropy (ME) Markov Model [1], Conditional Random Field (CRF) [2, 3] and Hidden Markov Model [1]. However, these traditional models require a large amount of linguistic knowledge to manually construct features, and therefore the applications are limited. Unlike, the Deep Learning utilizes unsupervised data effectively, avoid cumbersome artificial feature extraction, and has excellent generalization ability. Also, the Deep Learning models the data in multiple levels, obtaining the hierarchical structure of data features and the distributed representation of data.

With the improvement of hardware, large-scale computing has become possible. The development of deep learning and word embedding technology makes neural network (NN) a popular choice for natural language processing, which greatly reduces the workload of feature engineering, and more and more scholars apply NN to solve Chinese word segmentation. At present, the traditional word segmentation methods require a large amount of linguistic knowledge to manually construct features and the results rely on the size of the dictionary. As a result, the applicability of these methods is strictly limited. Compared with the traditional methods, the neural network effectively utilizes unsupervised learning data and avoids cumbersome artificial feature extracting work, providing better generalization...
ability. The characteristics and the structure of the data are obtained through a multi-level model. The advantages are as follows: (1) By constructing the NN model, the features are automatically learned. (2) A large amount of unmarked data is easier to obtain than the labeled data does. (3) There are many models that can be used, and better results can be achieved. The NN has been used in many methods, such as direct use of NN for Chinese word segmentation [4], a Gated Recursive Neural Network (GRNN) with an adaptive gate [5-7] and a Long Short-Term Memory Neural Networks [8, 9] (LSTM) for extracting features [10, 11]. In recent years, there are also some methods such as direct use of bidirectional long-term and short-term memory model (BI-LSTM) networks [12-14], the use of BI-LSTM networks combined with CRF [15], the use of BI-LSTM networks combined with CRF and Attention [16], and direct use of a CNN network [1].

Traditional methods often solve these problems separately, ignoring the relationship between these problems. Using deep learning method, these problems can be handled simultaneously by building a model for multiple features extracting. And multitasks learning methods can be used to model the relevance of features for better performance. However, large-scale deep learning network takes a long time to run, and is constructed with a complex model, which requires advanced support from computer hardware. Therefore, this paper uses a small-scale neural network, which takes the advantages of the neural network, ensures fast training, and requires less hardware support. CNN is used to extract and recombine features to solve the problems above: the input of BI_LSTM model is traditional Chinese character vector, the features learned is limited, and the learning efficiency is low. As for the BI_LSTM, this is suitable for Chinese NLP tasks. Combing these two models and adds the weight sharing method is used to classify each Chinese character. Taking the whole sentence as an input, the model uses the CNN network to extract and recombine the multi-dimensional features of each Chinese character, solving the problem that the BI_LSTM network can only learn limited features. And then the BI_LSTM network is used for sequence processing to obtain contextual relationships, which solves the sequence processing shortcoming from the CNN network. Finally, the weight sharing method is used to classify with SoftMax, thus the classification of each words in a whole sentence becomes the output. For this algorithm both the advantages of the BI_LSTM network in sequence processing and the CNN network in feature extraction are combined. The model is more efficient than the traditional BI_LSTM network, and the accuracy rate is increased to 98%.

2. Model Design

2.1. Model Overview

In order to solve the disadvantages of the BI_LSTM network and the CNN network mentioned above, the CNN + BI_LSTM model is proposed to process Chinese word segmentation in this paper shown in figure 1. In figure 1, from left to right: (1) Put the whole sentence into the model and performs the Embedding processing. (2) The output of the step (1) is put into the CNN network for Chinese character vector feature extraction. (3) Use the BI_LSMT network to obtain context associations. (4) Use weight sharing method to finish classification.

![Figure 1. CNN+BI_LSTM model.](image-url)
The model uses the matrix formed by the whole sentence as the input. To select the most suitable size of the convolution core, the comparative experiments are carried out. Then, the CNN network is used for feature extraction to obtain multi-dimensional features of the word vector, which will be sent to the BI_LSTM network for learning. Finally, the full connection of weight sharing is used to classify each word. The super-parameter value is adjusted several times to find more and more accurate results compared with other models. The advantages of CNN in feature extraction and the advantages of BI_LSTM in coping with sequence are combined together to improve the accuracy of the output. Each character’s label in the whole sentence is output at the same time by adopting the full connection of weight sharing.

2.2. Labeling Method
The dataset in this paper is composed by a large amount of data in different fields such as novels, news data, microblog, BBS, and product evaluations. It contains a large number of different Chinese characters and words, which ensures the applicability of the model in variable articles.

In the sequence labeling learning method which is based on word symbol learning [18, 19], the word segmentation symbolizes each word with a unique label. The method uses a four-word symbol method in which each character corresponds to one of the four symbols, namely SBME (single word, begin word, middle word and end word, as shown in table 1), indicating how the words are segmented.

| 中国 人 | / | 勤 劳 | / | 勇 敢 | / | , |
|--------|---|-------|---|-------|---|----|
| B      | M | E     | B | E     | B | E | S |

Note: S = single word, B = begin word, M = middle word, E = end word.

2.3. Embedding Layer
In NLP tasks, converting text into digital representation is an indispensable process. In traditional methods such as One-Hot coding, the core idea is to create a unique sequence number for each word for its position in the dictionary. This method will produce very long vectors when the text includes a lot of different words. Only the number corresponding to the position of the word is 1, and the rest are all 0s, resulting in a sparse word vector which is not easy to learn. In contrast, the Distributed Representation [20] is trained to learn the similarity between the words to obtain the vector distance between them (such as Euclidean distance). Its quality is affected by the size and the effectiveness of the training set. The small-sized or ineffective training set may lead to errors in the processing of word segmentation. This paper uses One-Hot encoding method and maps it into a low-dimensional space using the Embedding layer, which solves the problem of sparse matrix and gives the parameters a large learning space.

In the Embedding layer the matrix is defined as shown in equation (1). The matrix $A$ is $32*d$, the matrix $A$ has at most 32 rows mean the sentence length and the $d$ is the dimension of the word vector.

$$A = \begin{bmatrix}
a_{1,1} & a_{1,2} & \cdots & a_{1,d} \\
a_{2,1} & a_{2,2} & \cdots & a_{2,d} \\
\cdots & \cdots & \cdots & \cdots \\
a_{32,1} & a_{32,2} & \cdots & a_{32,d}
\end{bmatrix} = \begin{bmatrix}
\cdots a_1 \cdots \\
\cdots a_2 \cdots \\
\cdots \cdots \\
\cdots a_{32} \cdots
\end{bmatrix}$$

The purpose is to use the whole sentence as input to facilitate the BI_LSTM network to obtain contextual features recombining by CNN and to ensure that the final output is the label of each word in a whole sentence. After the Embedding layer, each sentence is converted into a matrix as the input of the simple CNN to extract the new feature.
2.4. Simple CNN Layer Design

The CNN is widely used in the field of visual imaging. With the development of word vector and deep learning, many NLP tasks directions can also be solved by CNN, such as sentences and article classification, sentiment analysis, and texts summary. Further more, the CNN is not limited by the specific language or the specified syntactic grammar, and has a wide range of practicalities. Similar to the traditional image domain, NLP also contains a combination of convolutional and pooled layers, compared with the smoothness required in the field of computer vision, and the word segmentation process needs to input as much as possible characteristics of each word. The blind use of the pooling process may probably cause feature loss, reducing the accuracy. In the experiment, the pooling comparison was carried out in the small-scale model, and the results were consistent with the expectations. In this task, the pooling not only failed to improve the accuracy, but also reduced the training speed of the model, and therefore, the maximum pooling and the average pooling will lose more features. Although K-max pooling can retain a part of its feature, the result is not as good as the one without the pool layer. Therefore, the convolutional network in this paper does not use the pooling layer operation. Comparisons of various pooling methods are shown in table 2.

| Accuracy         |
|------------------|
| Max-pooling      |
| 71.05%           |
| Average-pooling  |
| 74.68%           |
| 4-max-pooling    |
| 78.36%           |
| No-pooling       |
| 88.28%           |

The second layer is a convolutional layer, in which the word vector matrix of each sentence runs through the two-dimensional convolution (as shown in equation (2)). With the advantages of CNN network for feature extraction, the feature of each Chinese character vector is extracted by the convolution kernel of (1, word_size). The multi-channel-output results of the convolutional layers are recombined into new multi-dimensional features for each Chinese character. The purpose is to replace the original Chinese character vector with the new multi-dimensional feature vector. Compared with the original Chinese character vector, the new feature vector contains more dimensional word features, so that the new Chinese character feature can be learned by subsequent networks.

Let the number of rows and columns of the matrix be $M_r$, $M_c$ and $N_r$, $N_c$ respectively, as shown in equation (2)

$$C(s,t) = \sum_{m=0}^{M_r-1} \sum_{n=0}^{M_c-1} A(m,n) \cdot B(s-m,t-n)$$

where $0 \leq s < M_r + N_r - 1$, $0 \leq t < M_c + N_c - 1$.

The choice of the size of the convolution kernel in the CNN layer determines the way where the features are extracted, having a greater impact on subsequent word segmentation. This experiment designed a comparing experiment between a single-layer CNN network and a two-layer CNN network, using (1, word_size), (2, word_size) and (2, 1)+(1, word_size) double-layer CNN networks. The purpose is to compare the convolution kernel for the feature extraction of the Chinese character vector. The comparison test model is shown in figure 2. The comparison results are shown in table 3 (the activation function is Relu, $\beta$ is 1). The single-layer convolution kernel directly adopts the extraction of features in one or two Chinese character vector features. The two-layer convolution kernel performs feature extraction in a single-dimension of the two Chinese characters vector to obtain the characteristic relationship between two Chinese characters, using (1, word_size) dimensional compression to get the connection between phrases.
Figure 2. Convolution kernel selection comparison model.

Table 3. Comparison of multi-convolution kernels and multilayer convolution layers test results.

| SBME       | Precision | Recall  | F1       | Accuracy |
|------------|-----------|---------|----------|----------|
| (1, word_size) | 88.28%    | 88.37%  | 88.32%   | 88.28%   |
| (2, word_size) | 85.36%    | 85.26%  | 85.30%   | 85.26%   |
| 2-conv-layers | 87.67%    | 87.01%  | 87.33%   | 87.93%   |

It can be seen from table 3 that different convolution kernels achieve different effects of word segmentation with standard stability and accuracy, but slightly differences in accuracy. Contrary to the expected result, the convolution between the two Chinese characters did not get good results. Instead, the Chinese character self-convolution achieved good results through the fully connected layer. Taking the sentence as the unit input, the full connection layer breaks the spatial sentence structure. This achieves the learning effect between two Chinese characters, and result a high accuracy. The convolution kernel (2, word_size) and the two-layer convolutional layer model are used to extract features between two Chinese characters. During the BI_LSTM model training, the relationship between the time steps is no longer the one between two Chinese characters, but the one between two phrases, contrary to the purpose of the word segmentation, leading to a decline in accuracy. If the four-word-labeling is replaced by two-word-labeling, the accuracy of the word segmentation is improved, but the variety of classification is declined, leading to relatively vague word segmentation. It is found that the results of the two-layer CNN network are not as effective as those with single-layer CNN network, though the model fails to achieve the expected effect on the feature extraction. After the experiment above, the convolution kernel of (1, word_size) is finally selected.

2.5. BI_LSTM Layer
The third layer is the BI_LSTM layer, and the new feature vector obtained from the previous layer is
taken as input. Compared with the traditional BI_LSTM network, the Chinese character vector is
directly used as input. The model algorithm uses the multi-dimensional feature of the Chinese
character vector as input. Learning based on the characteristics of the Chinese character vector of the
multi-dimensional feature, BI_LSTM network improves the accuracy of the output and the efficiency
of learning. This layer mainly uses the advantages of BI_LSTM network to deal with timing problems
and uses the extracted new feature vectors, solving the weakness of CNN network and increasing the
learning ability of the network. The BI_LSTM model is an improved LSTM model generated by a
traditional RNN model with a two-way time dimension added to the model. It can solve the problem
that the traditional RNN model is prone to gradient vanishing or gradient explosion when the sequence
is long, and can also remember and merge the past and future information through the forward and
backward units.

2.6. Weight Sharing and Classification
The fully connection layer of the weight sharing first performs a weight-sharing in the full connection
for each output, and then performs a SoftMax classification in the full-connected result of each output
to obtain a corresponding label. The core is that the sequence of length 32 of the BI_LSTM network
output is independent of each other and shares the weight of the full connection operation. The output
vector of each BI_LSTM network achieves an independent classification effect, thereby achieving a
one-to-many effect. During the classification, the SoftMax enters the entire sentence and produces the
classification output for each Chinese character of the whole sentence. This layer ensures that the
multiple targets are independently classified in time sequence. The weight sharing ensures the
relevance of words in the sentence during training. Finally, the SoftMax function is used to classify the
corresponding labels (as show in figure 3).

![Figure 3. Weight share.](image)

In order to prevent the over-fitting problem in the model training process, this paper uses the
Dropout [21] technique. The core is to randomly remove a certain proportion of p (Dropout ratio)
neurons and their corresponding input and output weights during model training. Adding the Dropout
method to the fully connected layer of the CNN_LSTM model reduces the error rate and improves
system performance.
3. Experimental Results

3.1. Dataset
The dataset uses a large number of novels, news data, blog, BBS, commodity evaluation and other datasets to learn and test, covering a large number of Chinese characters and words. The data statistics table is shown in Table 4.

|                          | Big Dataset | PKU    | MSR    |
|--------------------------|-------------|--------|--------|
| Number of phrases (no repetition) | 200,150     | 55,303 | 88,119 |
| Number of Chinese characters (no repetition) | 7,817       | 4,698  | 5,167  |
| Total number of phrases   | 8,224,014   | 1,109,947 | 2,368,391 |
| Total number of Chinese characters | 20,683,689 | 1,826,488 | 4,050,469 |

For the division of datasets, the size of the datasets is huge. The 8:1:1 segmentation ratio is used to divide the datasets into train-set, valid-set and the test-set, which is training-set: valid-set: test_set = 8:1:1. The data of the training-set, valid-set and the test-set is shuffled before used.

3.2. Hyper Parameter
The experimental environment is sustained by the CentOS 7 system, Python 3.6, and Keras 2.2.4.

The hyper-parameters of the neural network are crucial to the pros and cons of the neural network model. After repeated debugging, we selected the hyper-parameters in Table 5 as the hyper-parameters used in the final model of the experiment.

| Parameter          | Value   |
|--------------------|---------|
| Epoch              | 5       |
| Batch-Size         | 256     |
| Learn-Rate         | 0.001   |
| Word-Size          | 128     |
| In-Channel         | 1       |
| Out-Channel        | 64      |
| Activation function| Relu    |
| BI_LSTM layer      | 1       |
| Optimization       | Adam    |

In the training process of neural network, the Epochs number reflects the convergence speed of the model. Therefore, we draw the accuracy corresponding to different Epochs values. The experimental results on the datasets are shown in figure 4, so the final selection epochs is 5.

3.3. Results Evaluation
The experimental evaluation used Precision, Recall, F1 value, and Accuracy as the criteria for evaluating the model. (F1: \( \beta=1 \). After selecting the convolution kernel size, CNN+BI_LSTM is compared to the model developed in the last two years, as shown in tables 6 and 7.
Figure 4. Epoch curve.

Table 6. Experimental comparison results.

| Score                  | Precision | Recall | F1      | Accuracy |
|------------------------|-----------|--------|---------|----------|
| CNN+BI_LSTM            | 98.03%    | 97.97% | 97.99%  | 98.00%   |
| BI_LSTM                | /         | /      | /       | 97.00%   |
| BI_LSTM_CRF            | /         | /      | /       | 96.32%   |
| PCNN                   | /         | /      | /       | 96.48%   |
| BI_LSTM_CRF_Attention  | /         | /      | /       | 97.28%   |

Table 7. Result in MSR, PKU Dataset.

| SBME                  | MSR |       | PKU |       |
|-----------------------|-----|-------|-----|-------|
|                       | P   | R     | F1  | A     |
|                       |     |       |     |       |
| CNN+BI_LSTM [8]       | 97.88% | 97.90% | 97.89% | 97.90% | 97.00% | 96.88% | 96.94% | 96.98% |
| BI_LSTM [8]           | %   | %     | %   | %     | %   | %     | %     | %     |
|                       | 96.20% | 95.20% | 95.70% | / | 95.30% | 94.60% | 94.90% | / |
| BI_LSTM_CRF [15]      | 97.10% | 96.00% | 96.50% | / | 96.20% | 95.60% | 95.90% | / |
| PCNN [17]             | %   | %     | %   | %     | %   | %     | %     | %     |
|                       | 95.17% | 95.07% | 95.09% | 95.07% | 94.60% | 94.44% | 94.46% | 94.44% |
| BI_LSTM_CRF_Attention | %   | %     | %   | %     | %   | %     | %     | %     |
|                       | 97.50% | 97.60% | 97.50% | / | 96.60% | 96.40% | 96.50% | / |

As expected before the experiment, the use of CNN network, extracting the features of Chinese character vectors and combining them into new Chinese character vectors, is improved compared with the traditional BI_LSTM network. Traditional BI_LSTM network contains the only the vector of each Chinese character as its input to calculate. The input designed in this paper is a multi-feature combination of Chinese character vectors features, greatly increasing the characteristics of network learned. This algorithm not only combines the advantages of CNN in feature extraction, but also takes advantage of the LSTM network to handle timing issues. Compared with the traditional BI_LSTM network, the accuracy and convergence speed is increased. As a result, the accuracy of the model reaches 98%.

4. Conclusion and Expectation
A new algorithm combined with the simple CNN network, the BI_LSTM network, and the weight-sharing full connection proposed in this paper. The feasibility of the algorithm is proved by the
large-scale datasets based on the multi-domain. The model completes the word segmentation task efficiently. The convergence speed and accuracy is raised. It utilizes not only the advantages of feature extraction of CNN network, but also the advantages of BI_LSTM network in timing. By extracting the multidimensional features of the word vector, the effect of training is improved. The weight sharing enables the model to simultaneous classification the sentence. Due to laboratory conditions and personal time, there is no deep exploration and exploration, and the algorithm has a relatively large space for exploration.

Reference
[1] Le Y L and Chang B B 2010 Chinese word segmentation method based on maximum interval Markov network model Chinese Journal of Information 24 (1) 8-14.
[2] Peng F, Feng F and McCallum A 2004 Chinese segmentation and new word detection using conditional random fields Proceedings of the 20th International Conference on Computational Linguistics (Association for Computational Linguistics) p 562.
[3] Tseng H, Chang P and Andrew G 2005 A conditional random field word segmenter for sighan bakeoff 2005 Proceedings of the fourth SIGHAN workshop on Chinese language Processing Asian Federation of Natural Language Processing 105-3027.
[4] Zheng X, Chen H and Xu T 2013 Deep learning for Chinese word segmentation and POS tagging Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP) Association for Computational Linguistics (ACL) 647-657.
[5] Zhou G B, Wu J, Zhang C L, et al. 2016 Minimal gated unit for recurrent neural networks International Journal of Automation and Computing 13 (3) 226-234.
[6] Chung J, Gulcehre C, Cho K H, et al. 2014 Empirical evaluation of gated recurrent neural networks on sequence modeling arXiv preprint arXiv 1412.3555.
[7] Chen X, Qiu X, Zhu C, et al. 2015 Gated recursive neural network for Chinese word segmentation Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Association for Computational Linguistics (ACL)) pp 1744-1753.
[8] Chen X, Qiu X, Zhu C, et al. 2015 Long short-term memory neural networks for Chinese word segmentation Conference on Empirical Methods in Natural Language Processing pp 1197-1206.
[9] Chen X, Qiu X, Zhu C, et al. 2015 Long short-term memory neural networks for Chinese word segmentation Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (Association for Computational Linguistics (ACL)) pp 1197-1206.
[10] Xu J and Sun X 2016 Dependency-based gated recursive neural network for Chinese word segmentation Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Association for Computational Linguistics (ACL)) pp 567-572.
[11] Zhang W, Zhang H, Yang L and Xun E 2019 Multi-grained Chinese word segmentation with lattice-LSTM Journal of Chinese Information Processing (01) 18-24.
[12] Xu W, Chen W and Liu T 2019 Chinese word segmentation by integration of handcraft-feature and Bi-LSTM Intelligent Computer and Applications (01) 169-172.
[13] Zhang Y and Qian J 2018 Text sentiment analysis based on LSTM structures Computer Engineering & Software 39 (1) 116-120.
[14] Yao Y and Huang Z 2016 Bi-directional LSTM recurrent neural network for Chinese word segmentation International Conference on Neural In-formation Processing pp 345-353.
[15] Yao M, Li H, Lü H and Yao D 2019 Chinese word segmentation method based on BI_LSTM_CRF neural network for sequence labeling Modern Electronics Technique (01) 95-99.
[16] Huang D and Guo Y 2018 BI-LSTM-CRF Chinese word segmentation model with attention mechanism Computer Engineering & Software 39 (10) 260-266.
[17] Tu W, Yuan Z and Yu K 2019 Chinese word segmentation based on pool-free layered
convolutional neural network Computer Engineering and Applications 1-8.

[18] Xue N 2003 Chinese word segmentation as character tagging International Journal of Computational Linguistics & Chinese Language Processing-Special Issue on Word Formation and Chinese Language Processing 8 (1) 29-48.

[19] Ng H T and Low J K 2004 Chinese part-of-speech tagging: One-at-a-time or all-at-once? word-based or character-based? Proceedings of the Conference on Empirical Methods in Natural Language Processing (Association for Computational Linguistics (ACL)) pp 277-284.

[20] Hinton G E 1986 Learning distributed representations of concepts Proceedings of the Eighth Annual Conference of the Cognitive Science Society (Lawrence Erlbaum Associates) p 12.

[21] Srivastava N, Hinton G, Krizhevsky A, et al. 2014 Dropout: A simple way to prevent neural networks from overfitting Journal of Machine Learning Research 15 (1) 1929-1958.