A Sentence Segmentation Method for Ancient Chinese Texts Based on NNLM

Boli Wang[1], Xiaodong Shi[1,2,3] (✉), Zhixing Tan[1], Yidong Chen[1], Weili Wang[1]

[1] Department of Cognitive Science, Xiamen University, Xiamen 361005, China
[2] Collaborative Innovation Center for Peaceful Development of Cross-Strait Relations, Xiamen University, Xiamen 361005, China
[3] Fujian Province Key Laboratory for Brain-inspired Computing, Xiamen University, Xiamen 361005, China
mandel@xmu.edu.cn

Abstract. Most of ancient Chinese texts have no punctuations or segmentation of sentences. Recent researches on automatic ancient Chinese sentence segmentation usually resorted to sequence labelling models and utilized small data sets. In this paper, we propose a sentence segmentation method for ancient Chinese texts based on neural network language models. Experiments on large-scale corpora indicate that our method is effective and achieves a comparable result to the traditional CRF model. Implementing sentence length penalty, using larger Simplified Chinese corpora, or dividing corpora by ages can further improve performance of our model.

Keywords: Ancient Chinese · Sentence segmentation · Neural network language model

1 Introduction

In ancient Chinese texts, characters are written or printed one by one without any punctuations or extra spaces to denote boundaries of words or sentences. To make texts more readable, readers must use their language expertise and understanding of context to mark the end of clauses and sentences by themselves, which is called Judou (句读, whose literal meaning is division of sentences and clauses). Because Judou is time-consuming, only a small amount of digitalized ancient Chinese texts have been manually segmented so far. Those unsegmented texts are hard to be understood by modern laymen and also difficult to process by computer programs.

Therefore, some researches apply NLP methods to automatic sentence segmentation of ancient Chinese texts. Popular methods [1,2,3] regard sentence segmentation as a sequence labeling task and exploit classical statistical models like conditional random fields (CRF) [4]. These models use handcrafted and fixed features, including characters in local context and their properties.

Intuitively, understanding of meanings in local and global context is indispensable for accurate sentence segmentation. Therefore, we regard sentence segmentation as a

M. Dong et al. (Eds.): CLSW 2016, LNAI 10085, pp. 387–396, 2016
DOI: 10.1007/978-3-319-49508-8_36

sequence generation task and employ the new neural network language model (NNLM), which introduces semantic information into the model. Giving a paragraph of unsegmented ancient Chinese text, we use a segmented ancient Chinse language model to generate punctuated text. In this paper, we only generate one type of punctuation mark (i.e. "。", which denotes the end of clauses). But our model can be extended to more complex punctuation sets.

Experimental results show that our model is comparable to the state-of-the-art CRF-based model. We also implement our model as a web service, which is available in http://jf2.cloudtranslation.cc/dj.html.

2 Related Work

2.1 Sentence Segmentation Method for Ancient Chinese Texts

[5] first proposed an n-gram model with a smoothing algorithm for ancient Chinese sentence segmentation. [6] handled sentence segmentation with a pattern-based string replacement approach. The precision of these early researches is rather low.

Later works treated sentence segmentation as a sequence labeling problem and tried CRF model with handcrafted feature templates. [1] used characters in local context. [2] implemented mutual information and T-test difference. [3] employed phonetic information, including modern Mandarin phonetic symbols Hanyu Pinyin (汉语拼音), and ancient Chinese phonetic symbols Fanqie (反切) and Guangyun (广韵). Experimental results show that CRF-based models are effective in ancient Chinese sentence segmentation. However, these methods rely mainly on handcrafted and limited features and are unable to utilize semantic information.

These existing researches trained and tested their models only on small-scale ancient Chinese corpora. Besides, there is no sentence segmentation toolkit or online service for ancient Chinese texts available yet.

2.2 Neural Network Language Model

Different from traditional statistical language model, neural network language models use distributed representation to represent semantic information of context [7]. [8] first introduced word embedding to language modeling. Words in context window are first projected to vectors and then feed into a neural network to estimate the conditional probability of next word. [9] proposed recurrent neural network based language model (RNNLM) which is capable to utilize information from full context. RNNLM is now a typical application of deep learning in natural language processing.

Note that the sentence segmentation model proposed in this paper is compatible with any kind of NNLM, despite the fact that we use RNNLM in our current implementation.

3 Our Approach

3.1 Sentence Segmentation Model

We regard ancient Chinese sentence segmentation as a sequence generation problem, which can be illustrated by the following formula:

$$\hat{Y} = \underset{Y,Y \in U(X)}{\operatorname{argmax}} P(Y) \tag{1}$$

where X is the input character sequence, i.e. unsegmented text; Y is the corresponding segmented text with extra punctuation marks added; $U(X)$ is a set of all possible segmentation results of X. For instance, giving $X =$ "三人行", then $U(X) =$ {"三人行","三。人行","三人。行","三。人。行"}.

Given a segmented sequence Y, we use language model to score the segmentation,

$$P(Y) = P_{lm}(Y) \tag{2}$$

where $P_{lm}(Y)$ is the joint probability of sequence Y estimated by a neural network language model.

However, language models prefer short sequences. Initial experimental results also show that this simple model tends to add a small number of punctuation marks. To address this issue, we introduced length penalty into our model:

$$P(Y) = P_{lm}(Y) + \lambda Len(Y) \tag{3}$$

where $Len(Y)$ is the length of sequence Y; λ is the weight of length penalty. Meanwhile, to make length penalty and probability of language model comparable, we transform $P_{lm}(Y)$ by taking its logarithm.

Given an unsegmented ancient Chinese paragraph, we use the beam search algorithm to find the optimal segmentation. In decoding, we keep N-best candidates temporarily for each step. We set N=15 in our experiments.

Furthermore, we use a heuristic method to deal with headlines, where no punctuation marks should be added. We propose following rules to judge whether an extra punctuation mark should be added to the end of the given sequence or not:

- If any punctuation marks have been added in decoding result, a punctuation mark should be added to the end of the sequence (because it seems to be a common paragraph).
- If no punctuation mark has been added in decoding result, no punctuation mark should be added to the end of the sequence either (because it seems to be a headline).

3.2 Neural Network Language Model

In this paper, we use character level recurrent neural network language models. We train two types of language models on segmented ancient Chinese texts. The first one is character-level language model with six punctuation types (CLM6), which means the training corpus only contains six types of punctuation marks, i.e. "。", "？", "！", "，", "；", and "：", and other punctuation marks are all removed. For the second one, we replace all six types with a unified segmentation mark, i.e. "。" and call this CLM1.

Given a sequence $S = w_1, w_2, \dots, w_n$, RNNLM use a vector $\boldsymbol{x}_t$ to represent the semantic information of each character w_t. The conditional probability is estimated by a recurrent neural network as follow:

$$P(w_{t+1} = i | w_t \cdots w_1) = \frac{exp(\boldsymbol{W}_i \boldsymbol{h}_t)}{\sum_{j=1}^{K} exp(\boldsymbol{W}_j \boldsymbol{h}_t)} \tag{4}$$

where K is the size of the vocabulary; $\boldsymbol{W}_j$ is the j-th row of the weight matrix; and $\boldsymbol{h}_t$ is the encoded hidden state of RNN, which is regarded as compressed representation of current context.

At each step, RNN updates the hidden state,

$$\boldsymbol{h}_t = f(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}) \tag{5}$$

where $\boldsymbol{x}_t$ is the current input, i.e. the vector representation of current character w_t; $\boldsymbol{h}_{t-1}$ is the hidden state of the previous step; and f is a non-linear function. In this paper, we employ the new Gated Recurrent Unit (GRU) as the non-linear function f [10,11].

GRU updates $\boldsymbol{h}_t$ as follows:

$$\boldsymbol{r}_t = sigmoid(\boldsymbol{W}_r \boldsymbol{x}_t + \boldsymbol{U}_r \boldsymbol{h}_{t-1}) \tag{6}$$

$$\boldsymbol{z}_t = sigmoid(\boldsymbol{W}_z \boldsymbol{x}_t + \boldsymbol{U}_z \boldsymbol{h}_{t-1}) \tag{7}$$

$$\widetilde{\boldsymbol{h}}_t = tanh(\boldsymbol{W}_h \boldsymbol{x}_t + \boldsymbol{U}_h (\boldsymbol{r}_t \circ \boldsymbol{h}_{t-1})) \tag{8}$$

$$\boldsymbol{h}_t = \boldsymbol{z}_t \circ \boldsymbol{h}_{t-1} + (1 - \boldsymbol{z}_t) \circ \widetilde{\boldsymbol{h}}_t \tag{9}$$

where $\boldsymbol{r}_t$ is called reset gate; $\boldsymbol{z}_t$ is called update gate; $\boldsymbol{W}_r$, $\boldsymbol{W}_z$, $\boldsymbol{W}_h$, $\boldsymbol{U}_r$, $\boldsymbol{U}_z$, and $\boldsymbol{U}_h$ are weights; ∘ denotes the element-wise product of the vectors.

4 Experiments

4.1 Setup

Different from existing works, our experiments are based on large scale corpora. We extract manually punctuated or segmented ancient Chinese texts from the Superfection Traditional Chinese Corpus (STCC) [1] and construct a training set of 237 million Chinese characters. Besides, we also construct a development set and two test sets, using segmented ancient Chinese texts in Traditional Chinese extracted from Hanchi

[1] http://cloudtranslation.cc/corpus_tc.html

Database of Sinica Taiwan (HDST)[2] and 4HN website[3]. Table 1 shows the details of these datasets.

Table 1. Details of datasets.

Dataset	# of chars	Charset size	Source	Content
Training Set	237M	23905	STCC	
Dev. Set	0.01M	1890	HDST	Chap. 1 of *Yue Wei Cao Tang Bi Ji (阅微草堂笔记)*
Test Set 1	0.32M	6188	4HN	*Bin Tui Lu (宾退录)* *Chao Ye Qian Zai (朝野佥载)* *Nan Bu Xin Shu (南部新书)* *Chu Ci Bu Zhu (楚辞补注)* *Zhong Wu Ji Wen (中吴纪闻)*
Test Set 2	0.36M	5755	HDST&4HN	Chap. 2 to Chap. 24 of *Yue Wei Cao Tang Bi Ji (阅微草堂笔记)* *Jing Zhai Gu Jin Tou (敬斋古今黈)*

We calculate precision P, recall R, and $F1$ score for sentence segmentation evaluation.

$$P = \frac{TP}{TP+FP} \tag{10}$$

$$R = \frac{TP}{TP+FN} \tag{11}$$

$$F_1 = \frac{2 \cdot P \cdot R}{P+R} \tag{12}$$

where TP, i.e. true positive, is the number of correct segmentation tags the model outputs; FP, i.e. false positive, is the number of wrong segmentation tags the model outputs; FN, i.e. false negative, is the number of wrong non-segmentation tags the model outputs.

[2] http://hanchi.ihp.sinica.edu.tw/ihp/hanji.htm

[3] http://www.4hn.org/

We reproduce the CRF-based sentence segmentation method as a baseline using the open-source toolkit CRF++[4]. We use the same feature template and training algorithm as [1]. We set the minimal frequency of features to 3 and use the default setting of the toolkit for other hyper-parameters.

4.2 Parameter Selection

The weight of length penalty λ is a significant hyper-parameter of our model. We evaluate the segmentation performance of CLM6 with different λ on development set. The results are shown in Fig. 1. With λ growing up, the recall increases and the precision drops accordingly. We find that when setting $\lambda = 0.65$, CLM6 achieves the highest $F1$ score.

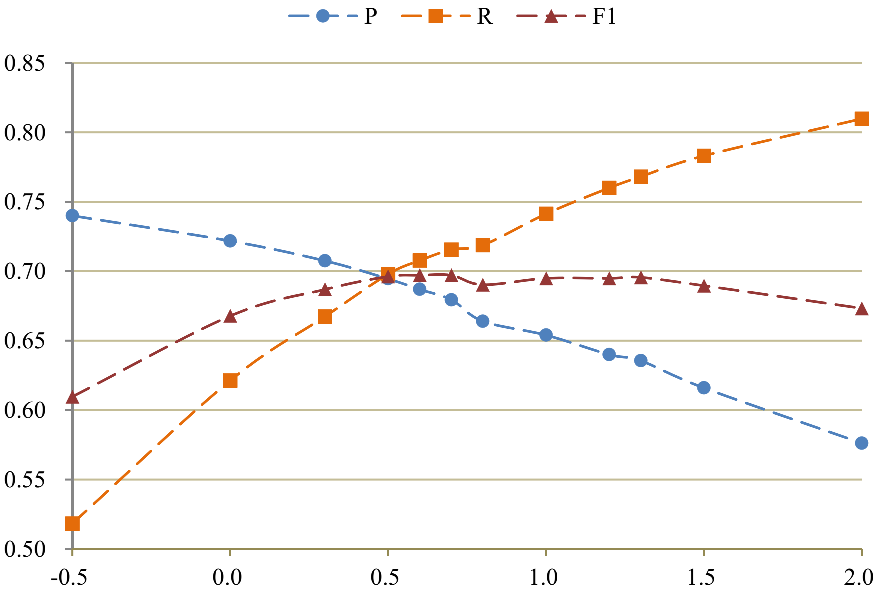

Fig. 1. Performances of CLM6 on different weights of length penalty λ.

4.3 Results

We compare our models, CLM6 and CLM1, with the baseline CRF-based model on two test sets. The experimental results are shown in Table 2. On precision, CLM6 outperforms CLM1 but cannot beat CRF-based model. CLM1 achieves higher recall

[4] http://taku910.github.io/crfpp/

than CLM6 and CRF baseline. On $F1$ score, both CLM1 and CLM6 cannot outperform CRF baseline, but are comparable.

Table 2. Experimental results on sentence segmentation.

	Test Set 1			Test Set 2		
	P	R	F1	P	R	F1
CLM6 (λ =0)	0.7822	0.5767	0.6639	0.7283	0.6364	0.6793
CLM6 (λ =0.65)	0.7492	0.6727	0.7089	0.6861	0.7205	0.7029
CLM1 (λ =0.65)	0.7212	**0.7325**	0.7268	0.6393	**0.7691**	0.6982
CRF	**0.8163**	0.6617	**0.7309**	**0.7856**	0.7100	**0.7459**

4.4 Experiments on Large-Scale Simplified Chinese Corpus

Actually, there are much more segmented ancient Chinese texts available in Simplified Chinese (SC) than those in Traditional Chinese (TC). And the total amount of SC characters is smaller than TC characters, which may help alleviate the problem of data sparsity. Therefore, we suspect that models trained on larger SC corpus should perform better on sentence segmentation.

We extract 629 million characters of ancient Chinese texts in SC from Superfection Corpus[5]. Furthermore, we use a public toolkit[6] to convert the previous 237M TC training set into SC and combine these two corpus into a larger SC training set. We use the same method and setups as CLM1 to train a new language model on this SC set. We name this larger SC language model CLM1-S.

When segmenting sentences, we first convert the given TC sequence into SC and then generate the segmented sequence using the same method mentioned in Section 3.1. Since characters in TC sequence and SC sequence are corresponding one to one, it's easy to transfer the segmentation marks to the original TC sequence.

The experimental results are shown in Table 3. CLM1-S outperforms CLM1 in both two test set and achieves much higher $F1$ scores. This proves that employing SC-to-TC conversion does help to alleviate the data sparsity problem and improves performance of ancient Chinese sentence segmentation. Furthermore, this time the performance is better than the CRF model.

[5] Although the scale of this SC corpus is more than 2 times larger than the TC corpus mentioned previously, the charset size of SC one is only 21697, that is smaller than the TC one, which confirms our intuition.

[6] http://jf.cloudtranslation.cc/

Table 3. Experimental results on large-scale Simplified Chinese corpus

	Test Set 1			Test Set 2		
	P	R	F1	P	R	F1
CLM1 (λ=0.65)	0.7212	0.7325	0.7268	0.6393	0.7691	0.6982
CLM1-S (λ=0.65)	**0.8243**	**0.7988**	**0.8113**	**0.7180**	**0.8339**	**0.7716**

4.5 Experiments on Fine-Grained Model.

Our training set contains ancient Chinese texts from different times. However, lexicon and grammar of ancient Chinese varied a lot during long ages. Intuitively, subdividing training corpus by ages may lead to more accurate language models and achieve better segmentation.

Referring to the standard of Academia Sinica Ancient Chinese Corpus[7], we divide our 237M TC training set into three subsets, namely Remote Ancient Chinese (上古汉语) (up to West Han Dynasty), Middle Ancient Chinese (中古汉语) (from East Han Dynasty to the Southern and Northern Dynasties), and Modern Ancient Chinese (近古汉语) (from Tang Dynasty). Since both Test Set 1 and Test Set 2 only contain Modern Ancient Chinese texts, we train a smaller language model on Modern Ancient Chinese subset (114M), using the same method and setup as CLM1. We call this fine-grained model CLM1-J.

Experimental results are shown in Table 4. CLM1-J outperform CLM1 on Test Set 2 but achieves a lower $F1$ score on Test Set 1. Considering that *Chu Ci Bu Zhu (楚辞补注)* in Test Set 1 contains lots of texts of *Chu Ci (楚辞)*, which actually belongs to Remote Ancient Chinese, we construct Test Set 3 with all texts in Test Set 1 except *Chu Ci Bu Zhu (楚辞补注)*. As shown in Table 4, CLM1-J achieves higher $F1$ score than CLM1 on Test Set 3.

To summarize, a fine-grained model trained on less than half of the corpus can even perform better in sentence segmentation.

Table 4. Experimental results on Fine-Grained Model

	Test Set 1			Test Set 2			Test Set 3		
	P	R	F1	P	R	F1	P	R	F1
CLM1 (λ=0.65)	**0.7212**	0.7325	**0.7268**	0.6393	0.7691	0.6982	**0.6496**	0.7478	0.6952
CLM1-J (λ=0.65)	0.6659	**0.7544**	0.7074	**0.6442**	**0.8267**	**0.7241**	0.6387	**0.8076**	**0.7133**

[7] http://app.sinica.edu.tw/cgi-bin/kiwi/akiwi/kiwi.sh

5 Conclusion

In this paper, we propose an ancient Chinese sentence segmentation method based on NNLM. Experimental results show that sequence generation model based on NNLM can achieve comparable segmentation results with traditional CRF-based models. Moreover, introducing length penalty to our model is effective to improve recall and F1 score of sentence segmentation.

Further experiments indicate that training datasets are important to improve performance of ancient Chinese sentence segmentation. Subdividing training corpus by ages or using larger SC corpus can lead to more effective NNLM and achieve better segmentation.

In further studies, we will try to implement CRF model to our sequence generation model to compensate the language models and boost segmentation performance. Moreover, inspired by experimental results in Section 4.4, we will try to normalize variant characters in ancient Chinese texts to further reduce data sparsity. Replacing variant characters in both training set and test set with the normalized ones should result in better segmentation.

Acknowledgments. The work described in this paper is supported by the Special Fund Project of Ministry of Education of China (Intelligent Conversion System from Simplified to Traditional Chinese Characters), National High-Tech R&D Program of China (No. 2012BAH14F03), the National Natural Science Foundation of China (No. 61573294 and No. 61303082) and the Research Fund for the Doctoral Program of Higher Education of China (No. 20130121110040).

References

1. Zhang, H., Wang, X., Yang J., Zhou, W.: Method of sentence segmentation and punctuating for ancient Chinese literatures based on cascaded CRF. Application Research of Computers, 26(9):3326–3329 (2009) (in Chinese)
2. Zhang, K., Xia, Y., Hang, Y. U.: CRF-based approach to sentence segmentation and punctuation for ancient Chinese prose. Journal of Tsinghua University, 49(10):1733–1736 (2009) (in Chinese)
3. Huang, H. H., Sun, C. T., Chen, H. H.: Classical Chinese sentence segmentation. In: Proceedings of CIPS-SIGHAN Joint Conference on Chinese Language Processing (2010)
4. Lafferty, J. D., McCallum, A., Pereira, F. C. N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of ICML (2001)
5. Chen, T., Chen, R., Pan, L., Li, H., Yu, Z.: Archaic Chinese punctuating sentences based on context N-gram model. Computer Engineering, 33(3), 192–193 (2007). (in Chinese)
6. Huang, J., Hou, H.: On sentence segmentation and punctuation model for ancient books on agriculture. Journal of Chinese Information Processing, 22(4):31–38 (2008) (in Chinese)
7. Hinton, G. E.: Learning distributed representations of concepts. In: Proceedings of CogSci (1986)
8. Bengio, Y., Ducharme, R., Vincent, P.: A Neural Probabilistic Language Model. In: Proceedings of NIPS (2001)
9. Mikolov, T., Karafiat, M., Burget, L., Cernockk, J. H., Khudanpur, S.: Recurrent neural network based language model. In: Proceedings of Interspeech (2010)

10. Cho, K., Merrienboer, B. V., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
11. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 (2014)