Proceedings of the Second Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA 2022), pages 169–173
Language Resources and Evaluation Conference (LREC 2022), Marseille, 25 June 2022

Data Augmentation for Low-resource Word Segmentation and POS Tagging of Ancient Chinese Texts

Yutong Shen*†, Jiahuan Li*†, Shujian Huang*†, Yi Zhou†, Xiaopeng Xie†, Qinxin Zhao†
* National Key Laboratory for Novel Software Technology, †Nanjing University, China
{shenyt, lijh, yizhou, xiaopengxie, QXZhao}@smail.nju.edu.cn,
huangsj@nju.edu.cn

Abstract

Automatic word segmentation and part-of-speech tagging of ancient books can help relevant researchers to study ancient texts. In recent years, pre-trained language models have achieved significant improvements on text processing tasks. *SikuRoberta* is a pre-trained language model specially designed for automatic analysis of ancient Chinese texts. Although *SikuRoberta* significantly boosts performance on WSG and POS tasks on ancient Chinese texts, the lack of labeled data still limits the performance of the model. In this paper, to alleviate the problem of insufficient training data, We define hybrid tags to integrate WSG and POS tasks and design Roberta-CRF model to predict tags for each Chinese characters. Moreover, We generate synthetic labeled data based on the LSTM language model. To further mine knowledge in *SikuRoberta*, we generate the synthetic unlabeled data based on the Masked LM. Experiments show that the performance of the model is improved with the synthetic data, indicating that the effectiveness of the data augmentation methods.

Keywords: Ancient texts, Word segmentation and POS tagging , Data augmentation

1. Introduction

Ancient Chinese books are precious cultural heritage, and of extremely high research value. In recent years, the protection and research of ancient Chinese books have attracted much attention, and the research is imminent. Some automatic analysis work of ancient Chinese books, such as word segmentation(WSG) and part-of-speech tagging(POS), can help relevant researchers to study ancient books.

Natural language processing technology is becoming more and more mature in recent years, among which, pre-trained language models(PLM) have achieved remarkable improvements in a lot of tasks, including word segmentation and part-of-speech tagging (Devlin et al., 2018; Liu et al., 2019). In order to better carry out the research of ancient books, Wang et al. (2022) proposed *SikuRoberta*, a masked language model trained on the large scale high-quality *Siku Quanshu* full text corpus. *SikuRoberta* significantly boosts performance on WSG and POS tasks on ancient Chinese texts.

However, the scarcity of training data and the expensive cost of manual annotation still limit the improvement of the model performance on WSG and POS tasks. Thus, how to obtain better model performance based on the existing *SikuRoberta* in the case of low resources is a problem that needs to be solved.

In this paper, we adopt a joint-tagging framework, designing hybrid tags to integrate WSG and POS tasks, to train an end-to-end network for WSG-POS task. We combine *SikuRoberta* and conditional random field(CRF) to predict tags for each Chinese characters. In addition, we use data augmentation methods to alleviate the problem of insufficient training data. We leverage DAGA (Ding et al., 2020) to generate synthetic labeled data(lstm-data) based on the LSTM autoregressive language model. To further mine knowledge in *SikuRoberta*, we generate the synthetic unlabeled data(unlabeled-data) based on the Masked LM. Then, we use the tagger model which is trained on real-data and lstm-data to label the unlabeled-data for generating mlm-data. Finally, based on real-data, lstm-data and mlm-data, we use dynamic weight sampling to balance various types of data to train the final model.

The experimental results show that the performance of the model is improved with synthetic data, which verifies the effectiveness of the data augmentation methods.

The paper is organized into 7 sections. We describe the structure of Roberta-CRF model in Section 2. Two data augmentation methods are elaborated in Section 3. Section 4 describes the flow of our entire system. Section 5 presents the experiments and some analysis of the results. We also report our final submitted results in Section 6. Finally, some conclusions are drawn in Section 7.

2. Roberta-CRF Model

Compared with the traditional pipeline method, jointly conducting WSG and POS can improve performance in both two tasks (Shi et al., 2010). Thus, We define hybrid tags and build an end-to-end network.

2.1. Hybrid Tags

There are 4 kinds of word segmentation labels 'B', 'M', 'E' and 'S', which represent the beginning of a word, the middle of a word, the end of a word and the single-character word, respectively. There are 22 kinds of parts-of-speech labels, including verbs(v), nouns(n), location(ns), person(nr), and so on.

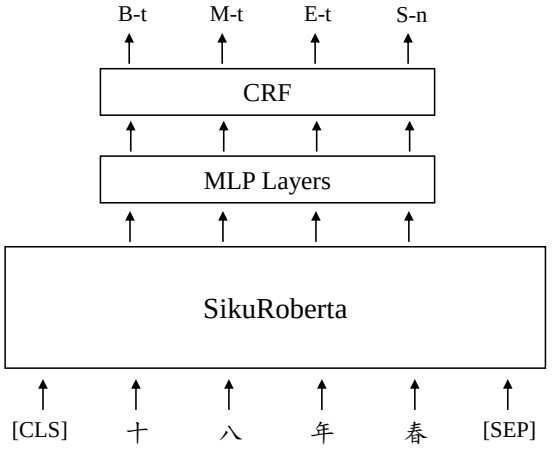

Figure 1: The structure of Roberta-CRF model

Each hybrid tag is composed of a word segmentation label and a part-of-speech label, and the two labels are connected by a connector '-'. For example, the two Chinese characters in the ancient Chinese word "春秋/n" will be marked as 'B-n' and ' E-n' respectively.

2.2. Model Structure

Formally, an ancient Chinese sentence S is sliced into n Chinese characters, denoted as $\{x_0, x_1, x_2, ..., x_{n-1}\}$, where n is the length of S. Our task is to get the corresponding tag sequence, denoted as $\{y_0, y_1, y_2, ..., y_{n-1}\}$.

We combine *SikuRoberta* and a CRF layer to form a Roberta-CRF model, whose structure is shown in Figure 1. *SikuRoberta* can produce the hidden states $H \in R^{n \times d}$ for the ancient Chinese sentence S, where d is the hidden layer size of the *SikuRoberta*. The hidden states H are fed into the two MLP layers to compute the emission scores for the CRF layer. The emission scores can be denoted as $Scores \in R^{n \times t}$.

$$Scores = W_2(W_1 H + b_1) + b_2 \quad (1)$$

where $W_1 \in R^{d \times d}$, $W_2 \in R^{d \times t}$, $b_1 \in R^d$, $b_2 \in R^t$ are the weight matrices and biases of the MLP layers respectively, and t is the number of hybrid tags.

CRF (Lafferty et al., 2001) has been widely recognized to be effective in sequence labeling tasks (Huang et al., 2015). As Eq. (2) shows, based on the emission scores, CRF calculates the tag sequence Y that maximizes the conditional probability using the Viterbi algorithm.

$$Y = \text{argmax}_y P(y|X) \quad (2)$$

where y is one of the all tag sequences of the same length as X. We update parameters of the entire network to minimize the loss function of CRF.

3. Data Augmentation

Data augmentation is one of the widely used methods in low-resource scenarios. To improve the performance of the Roberta-CRF model, we use two data augmentation methods, generating synthetic labeled and unlabeled data, respectively.

3.1. Labeled Data Generation

Ding et al. (2020) proposed a pseudo-data generation method for the sequence labeling task. We improve their method to generate pseudo label data for the WSG-POS task.

3.1.1. Modeling Text-Tag Hybrid Sequence

The model used to generate pseudo data is the LSTM (Shi et al., 2015) language model. Training dataset for this LM is the linearized labeled sentence. Linearizing the sentence is to insert the tag before the corresponding Chinese character. For example, our sentence is "十八年/t 春/n ， /w 白狄/nr 始/d /v 。/w", after linearization it is "B-t 十 M-t 八 E-t 年 S-n 春 S-w ， B-nr 白 E-nr 狄 S-d 始 S-v S-w 。"

We use the language model with the same structure as Ding et al. (2020). The only difference is that we set two independent embedding layers in our language model, one is tag embedding and another one is Chinese character embedding. The model structure can be seen in Figure 2

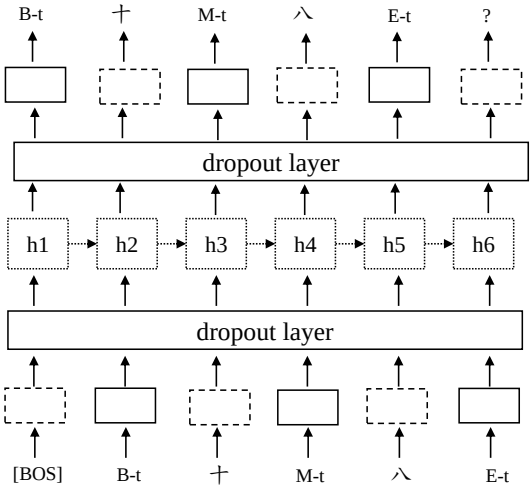

Figure 2: The structure of LSTM language model for linearized sentence.

We first feed the linearized sequence of characters $\{y_0, x_0, y_1, x_1, ..., y_{n-1}, x_{n-1}\}$ into the embedding layer to lookup the token embeddings $E = \{e_{t_0}, e_{c_0}, e_{t_1}, e_{c_1}, ..., e_{c_{n-1}}\}$.

$$e_{t_i} = \text{Embed}_t(y_i), \quad e_{c_i} = \text{Embed}_c(x_i) \quad (3)$$

where Embed_t and Embed_c are the embedding layers of tags and characters respectively. A dropout layer is applied to token embedding E to generate $D = dropout(E)$. Then, feed D into the single layer LSTM to produce hidden states $H = LSTM(D)$. Another dropout layer is applied to H to get $D' = dropout(H)$.

For the output layer, a linear and softmax layer are used to predict the next token in the sequence. Corresponding to the dictionary settings, there are two output layers, one is to generate the probability distribution P_t on the tags, and another one is to output the probability distribution P_c on the Chinese characters.

$$P_{t_i} = \text{Softmax}(W_c d'_{t_i} + b_c) \quad (4)$$

$$P_{c_i} = \text{Softmax}(W_t d'_{c_i} + b_t) \quad (5)$$

Where W_t and W_c are the weight matrices of tags and characters respectively.

Data Generation After training the LM, we can use it to generate synthetic labeled data for our task. During generation, only the '[BOS]' token is fed into LM, and the following tokens are sampled based on the probabilities computed by Eq. (4). and Eq. (5).

3.2. Unlabeled Data Generation

In our preliminary error analysis, we find most errors arises from words with POS tags of verbs, nouns, and rare words such as person name and locations. However, the language model in Section 3.1 can only generates sentences similar to the given training data, and may not generate novel aforementioned words. In contrast, *Sikuroberta*, trained on a lot of ancient Chinese texts, contains some ancient Chinese knowledge which can not be acquire from the given training data. To further mine knowledge in *SikuRoberta*, we generate new words or new characters using Masked LM based on *SikuRoberta*.

We randomly mask verbs, nouns, location and person in the training sentence with 20% probability, and ask SikuRoberta to fill the masked positions. When the masked positions are consecutive spans, we fill the span iteratively from left to right. This prevents the model to generate illegal words due the independent generation of each positions. For example, given a sentence "白狄始來", the process of generating masked words is shown in Figure 3

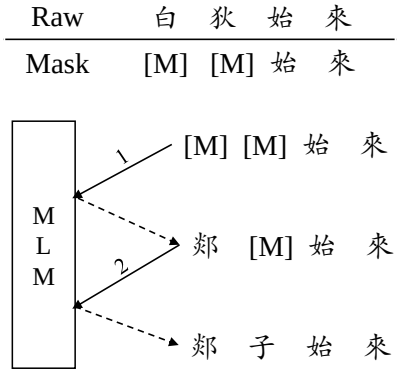

Figure 3: An illustration of the process of generating masked words iteratively

4. Implementation

We first train a tagger model based on the training data and labeled data generated by LSTM LM. Then, we use the tagger model to label the label unlabeled data generated by Masked LM. Finally, the final model is trained based on the three kinds of data.

4.1. Tagger Model

Based on LSTM LM, we generate 700k synthetic labeled data, denoted as D_{lstm}. Since the number of synthetic data is much larger than the training data, we adopt the method of dynamic weight sampling. The weights linearly changes according to the following equations:

$$W_{train_i} = \frac{i}{\text{N}} \times |D_{lstm}|$$

$$W_{lstm_i} = (1 - W_{r_i}) \times |D_{train}|$$

Where W_{train_i} and W_{lstm_i} are sampling weights of training and synthetic data, $|D_{train}|$ and $|D_{lstm}|$ are the number of training and synthetic data, $i = 0, 1, ...N-1$, N is the number of maximum epochs. It can be seen that the weight of sampling training data at the beginning is 0. With the epoch increasing, the weight of sampling training data becomes larger.

We save the 5 model checkpoints with the smallest loss during the training process and average their checkpoints as the tagger model.

4.2. Final Model

We generate 150k unlabeled data based on Masked LM. Then, we use the tagger model to label the unlabeled data, denoted these data as D_{mlm}.

Similarly, we train the final model based on the three kinds of data and adopt the method of dynamic weight sampling.

$$W_{train_i} = \frac{i}{N} \times |D_{mlm}| \times |D_{lstm}|$$

$$W_{mlm_i} = p \times (1 - W_{r_i}) \times |D_{train}| \times |D_{lstm}|$$

$$W_{lstm_i} = (1 - p) \times (1 - W_{r_i}) \times |D_{train}| \times |D_{mlm}|$$

where W_{mlm_i} and $|D_{mlm}|$ are the sampling weight and the number of D_{mlm}, and $i = 0, 1, ...N-1$. In our experiments, we set $p = 0.3$.

We also save the 5 model checkpoints with the smallest loss during the training process and average their checkpoints as the tagger model.

5. Experiments and Discussions

We randomly selected 1k data from the given training data as the in-domain test set and the rest as the training set. And we directly use Testb as the out-domain test set. Follow the settings in Section 4 to perform the experiments.

Model	In-domain	Out-domain
Roberta-CRF	92.14 / 84.49	86.94 / 75.09
Tagger	93.44 / 87.24	86.99 / 77.29
Final	**93.92 / 88.12**	**87.53 / 78.31**

Table 1: F1 scores of WSG and POS tasks

5.1. Results

We evaluate the model results using the F1 scores of the WSG and POS tasks. The results of the model on the test set are shown in Table 1.

It can be seen that the results of the Final model achieve the best performance for both in-domain and out-domain test sets.

Compared with the Roberta-CRF model, the F1 scores of final model results on the WSG and POS tasks are improved by 1.78 and 3.63 respectively for in-domain, and improved by 0.59 and 3.25 respectively for out-domain. It shows that data augmentation methods can enhance model performance.

For both in-domain and out-domain, the F1 scores of the Final model are also higher than Tagger model, indicating that the unlabeled data generated based on MLM can improve the model performance.

5.2. WSG: Analysis on Words of Different Frequencies

We divide the words appearing in the test set into frequent words, rare words and unknown words. Words that do not appear in the training set are unknown words. If a word appears less than 10 times in the training set, it is a rare word, otherwise it is a frequent word. We compute the accuracy rates of the three models on WSG task for different words.

		Roberta-CRF	Tagger	Final
In	fren(87.8%)	**0.969**	0.964	0.968
	rare(12.2%)	0.728	0.791	**0.807**
Out	fren(76.9%)	**0.919**	0.902	0.912
	rare(10.0%)	0.740	**0.752**	0.751
	unk(13.1%)	0.673	**0.717**	0.715

Table 2: The WSG correct rates of different words. Note that in-domian test set has no unknown words.

As shown in Table 2, for both in-domain and out-domain, the accuracy rates of the three models on frequent words are comparable. But for rare and unknown words, Roberta-CRF is the worst, its accuracy rates are 8% and 4.2% less than Final model respectively. This shows that rare and unknown words do affect performance of models, and adding pseudo data can significantly ease this problem.

5.3. POS: Analysis on Error Types

We also further analyze the results of the POS task. Since the part-of-speech tagging depends on the correct word segmentation, we only do the following statistics based on correctly segmented words. We count the types of POS errors as shown in Figure 4.

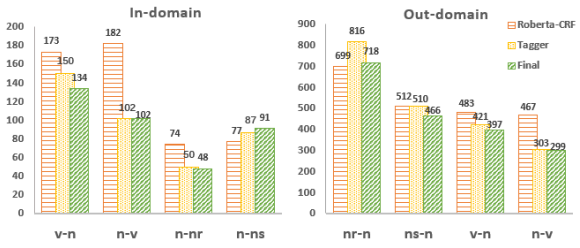

Figure 4: The types of POS errors. 'v-n' means that target label is 'v' but the output of model is 'n'.

in-domain It can be seen that the mutual misjudgment of nouns and verbs is the most common mistake. This is because a word in ancient Chinese often acts as both a verb and a noun, making it difficult for the model to distinguish between them.

out-domain However, for out-domain, there are many mistakes in labeling person as nouns. After using the data augmentation method, this error problems have not been effectively alleviated. But for other types of errors, pseudo data helps a lot.

6. Submitted System Results

For the final submitted results, we used ensemble learning to further improve the model performance. We randomly generate 10 sets of pseudo data, and then train 10 Final models respectively. Based on the 10 models, the results are obtained by voting.

on TestA with closed modality, our best F1 score of WSG is 94.81% and our F1 score of POS tagging is 89.87%. On TestB with closed modality, our best F1 score of WSG is 88.42% and our F1 score of POS tagging is 79.53%.

7. Conclusion

In this paper, we use the one-step approach, designing hybrid tags to integrate WSG and POS tasks, to train an end-to-end network for WSG-POS task. We combine *SikuRoberta* and conditional random field to predict tags for each Chinese characters.

Due to the limited training data, we use two data augmentation methods, generating synthetic labeled and unlabeled data, respectively. We refer to the DAGA to generate synthetic labeled data based on the LSTM language model. To further mine knowledge in *SikuRoberta*, we generate the synthetic unlabeled data based on the Masked LM. Finally we train the three kinds of data to obtain the final model.

The experimental results show that the performance of the model is improved after using data augmentation, which verifies the effectiveness of the data augmentation methods.

Appendix: Instances of Pseudo Data

An example of the pseudo sentences are shown in Figure 5.

Labeled data
Short 十八年/t 春/n ， /w 齊侯/nr 視/v 王/n 于/p 輔寘/ns 。/w **Long** 王/n 奉/v 以/p 朝/v ， /w 曰/v ：/w “/w 同王/n 不/d 能/v 久/a 先大夫/n ，/w 不/d 可/v 慎/v 也/y 。/w 若/c 問/v 諸/j 之 /r ， /w 不/d 可/v 處/v 。/w ‘/w 君子/n 無/v 精/n ， /w 不/d 可用/v 喪/v ， /w 守/v 備/n 而/c 興/v ， /w 不/d 賜/v 臣/n 以/p 定/v 之/r 。/w ”/w
Unlabeled data
Raw 十八年 ， 春 ， **白狄** 始 **來** 。 **Pseudo1** 十八年 ， 春 ， 郯子 始 來 。 **Pseudo2** 十八年 ， 春 ， 孫武 始 來 。 **Pseudo3** 十八年 ， 春 ， 白狄 始 興 。 **Pseudo4** 十八年 ， 春 ， 白狄 始 擾 。

Figure 5: Some instances of pseudo data

It can be seen that we can generate reasonable labeled sentences of varying lengths. In unlabeled sentence examples, the nouns and verbs in the original sentence are be randomly replaced with other nouns and verbs.

8. Bibliographical References

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ding, B., Liu, L., Bing, L., Kruengkrai, C., Nguyen, T. H., Joty, S., Si, L., and Miao, C. (2020). Daga: Data augmentation with a generation approach for low-resource tagging tasks. *arXiv preprint arXiv:2011.01549*.

Huang, Z., Xu, W., and Yu, K. (2015). Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Lafferty, J., McCallum, A., and Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Shi, M., Li, B., and Chen, X. (2010). Crf based research on a unified approach to word segmentation and pos tagging for pre-qin chinese. *Journal of Chinese Information Processing*, 2(24):39–45.

Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., and Woo, W.-c. (2015). Convolutional lstm network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 28.

Wang, C., Liu, Z., Zhu, J., Liu, H., Hu, Meditation, B., and Li. (2022). Sikubert and sikuroberta: Research on the construction and application of pre-trained model of “siku quanshu” for digital humanities. *http://kns.cnki.net/kcms/detail/44.1306.G2.20210819.2052.008.html*.