Neural Chinese Word Segmentation with Lexicon and Unlabeled Data via Posterior Regularization

Junxin Liu
Tsinghua University
Beijing, China
ljx16@mails.tsinghua.edu.cn

Fangzhao Wu
Microsoft Research Asia
Beijing, China
wufangzhao@gmail.com

Chuhan Wu
Tsinghua University
Beijing, China
wuch15@mails.tsinghua.edu.cn

Yongfeng Huang
Tsinghua University
Beijing, China
yfhuang@tsinghua.edu.cn

Xing Xie
Microsoft Research Asia
Beijing, China
Xing.Xie@microsoft.com

ABSTRACT

Chinese word segmentation (CWS) is very important for Chinese text processing. Existing methods for CWS usually rely on a large number of labeled sentences to train word segmentation models, which are expensive and time-consuming to annotate. Luckily, the unlabeled data is usually easy to collect and many high-quality Chinese lexicons are off-the-shelf, both of which can provide useful information for CWS. In this paper, we propose a neural approach for Chinese word segmentation which can exploit both lexicon and unlabeled data. Our approach is based on a variant of posterior regularization algorithm, and the unlabeled data and lexicon are incorporated into model training as indirect supervision by regularizing the prediction space of CWS models. Extensive experiments on multiple benchmark datasets in both in-domain and cross-domain scenarios validate the effectiveness of our approach.

CCS CONCEPTS

• **Computing methodologies** → **Natural language processing**; *Neural networks*; *Semi-supervised learning settings.*

KEYWORDS

Chinese word segmentation, Lexicon, Neural network

ACM Reference Format:
Junxin Liu, Fangzhao Wu, Chuhan Wu, Yongfeng Huang, and Xing Xie. 2019. Neural Chinese Word Segmentation with Lexicon and Unlabeled Data via Posterior Regularization. In *Proceedings of the 2019 World Wide Web Conference (WWW '19), May 13–17, 2019, San Francisco, CA, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3308558.3313437

1 INTRODUCTION

Chinese word segmentation (CWS) aims to segment Chinese sentence into words [7, 10, 22]. For example, "习近平常与特朗普通电话" is segmented into "习近平/常/与/特朗普/通/电话". Different from English texts where whitespace is used to separate words, there is no natural word delimiter in Chinese. Thus, CWS is very important for processing Chinese texts and is an essential step for many downstream tasks [3, 5, 15].

In recent years, neural network based methods have been widely used for CWS [1, 17, 23, 26]. Most of these methods model CWS as a sequence labeling problem [22, 30], and utilize neural networks to learn the hidden character features [2, 32]. For example, Chen et al. [2] used LSTM [8] to learn character features by capturing the global information of sentence. Peng and Dredze [18] proposed to use LSTM for character feature learning and CRF [9] for character label decoding. However, these methods usually rely on a large number of labeled sentences to train word segmentation models, which are expensive and time-consuming to annotate. Besides, these methods usually have difficulty in segmenting sentences with OOV (out of vocabulary) words or words that are rare in training data [27]. For example, if "习近平" and "特朗普" are OOV words in training data, then these methods will probably segment "习近平常与特朗普通电话" into "习近/平常/与/特朗/普通/电话".

Our work is motivated by following observations. First, the unlabeled Chinese sentences are usually easy to collect on a large scale and can provide useful information for Chinese word segmentation. For example, if the character sequences "习近平" and "特朗普" appear many times in unlabeled data with different contexts, then we can infer that they are probably Chinese words. Second, many high-quality Chinese lexicons have been built and can cover a large number of Chinese words. These lexicons can provide important information of whether a Chinese character sequence can be a valid Chinese word, which is useful for CWS. For example, if "习近平" and "特朗普" are included in a Chinese lexicon, then we can better segment aforementioned sentences. Thus, both unlabeled data and lexicons have the potential to improve the performance of CWS, especially on sentences with OOV and rare words.

In this paper, we propose a neural approach for Chinese word segmentation which can exploit the useful information in both Chinese lexicon and unlabeled data. More specifically, in our approach we propose a unified framework based on posterior regularization [6] to incorporate Chinese lexicon and unlabeled data as indirect supervision to regularize the prediction space of the neural CWS models. The neural CWS architecture used in our approach is CNN-CRF. The neural CWS model is trained based on both indirect supervision inferred from lexicon and unlabeled data and the direct supervision inferred from labeled sentences. Extensive experiments

WWW '19, May 13–17, 2019, San Francisco, CA, USA

ACM ISBN 978-1-4503-6674-8/19/05.
https://doi.org/10.1145/3308558.3313437

are conducted on multiple benchmark datasets in both in-domain and cross-domain scenarios. The experimental results show that our approach can effectively improve the performance of Chinese word segmentation, especially when training data is insufficient.

2 RELATED WORK

In recent years, neural network based methods have been widely used for Chinese word segmentation. These methods usually regard CWS as a character-level sequence labeling task. For example, Chen et al. [2] proposed to apply LSTM to Chinese word segmentation. They used LSTM to learn hidden character features by capturing the global context information of sentences. They also used a character window to capture local contexts for building character features. Peng et al, [18] used LSTM to learn the contextual character features and used CRF to jointly decode the labels of characters. These neural methods for CWS usually rely on a large number of labeled sentences for model training. When the labeled data is insufficient, the performance usually declines heavily [21, 27, 28].

Incorporating the useful information in unlabeled sentences and lexicons into CWS has attracted increasing attentions [11, 20, 23]. For example, Li et al. [11] proposed to utilize unlabeled sentences for CWS by using punctuation marks as implicit annotations. However, punctuation marks are sparse in Chinese texts, and the annotations of most characters cannot be obtained in this way. Sun et al. [20] proposed to extract statistics-based character features such as mutual information and accessor variety from unlabeled data, and use these features to improve CWS. Designing these handcrafted features needs a large amount of domain knowledge. Zhang et al. [27] proposed to incorporate lexicon into a neural CWS method based on LSTM-CRF architecture. They designed several handcrafted feature templates to extract additional character features using lexicon, and used another LSTM to learn character representations from these lexicon based features. Liu et al. [12] proposed to utilize lexicon for neural CWS via multi-task learning. They designed an auxiliary task of word classification to exploit lexicon information. Then they jointly trained CWS and word classification models using a multi-task learning framework. However, these methods cannot exploit the useful information in unlabeled data.

There are a few methods which can incorporate both lexicon and unlabeled data into Chinese word segmentation. For example, Liu et al. [13] proposed to build partially annotated data using unlabeled data and lexicons via many handcrafted rules. Then they trained CWS models based on both labeled and partially annotated data using CRF. However, designing these rules requires a lot of domain knowledge. Zhao et al. [31] used a similar way to build partially labeled data for CWS by combining unlabeled data and Chinese lexicons. They trained neural CWS models on both labeled and partially labeled data using a variant of LSTM. However, in these methods the partially annotated data is built simply by word matching without considering the contexts of sentences. Thus, the partially annotated data may contain heavy noise and is not suitable to use directly as training data. Different from these methods, in our approach the lexicon and unlabeled data are incorporated to provide indirect supervision via posterior regularization.

Posterior regularization algorithm [6] can incorporate additional knowledge into model training as constraints over the posterior distributions of models. For instance, Zhang et al. [24] applied posterior regularization to integrate various kinds of knowledge such as bilingual dictionary, phrase table and coverage penalty into neural machine translation and gained huge performance improvements. Our approach is motivated by posterior regularization, and we apply it to incorporate the useful information in Chinese lexicon and unlabeled data into neural Chinese word segmentation.

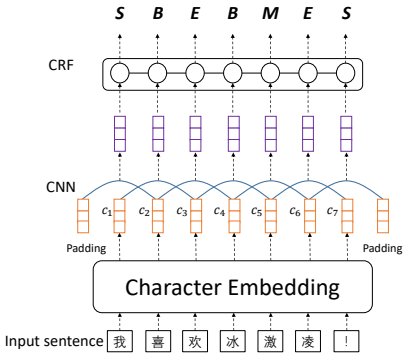


Figure 1: The CNN-CRF neural architecture for CWS.

3 OUR APPROACH

We first introduce the CNN-CRF neural architecture in our approach for Chinese word segmentation. Then we introduce our approach to incorporate lexicon and unlabeled data into neural CWS.

3.1 CNN-CRF Architecture

Following many previous works [14, 29], we model Chinese word segmentation as a character-level sequence labeling problem. For each character in a sentence, our model assigns a tag from a predefined tag set to it which indicates the position of this character in a word. The tag set used in our model is $\{B, M, E, S\}$, where B, M and E mean the beginning, middle and end position in a word respectively, and S means single character word.

We use CNN-CRF as the neural architecture for CWS which is illustrated in Fig. 1. The CNN-CRF architecture contains three layers. The first one is character embedding. Given a sentence $x = [c_1, c_2, ..., c_N]$, the character embedding layer will map each character to a low-dimensional dense vector. Here c_i is the i-th character in the sentence, and N is the sentence length. The output of this layer is $x = [\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_N]$, where $\mathbf{e}_i$ is the embedding of c_i.

The second layer is a CNN network. It is used to learn contextual representations of characters. The hidden feature of the i-th character learned by a convolutional kernel is:

$$h_i = f(\mathbf{w}^T \times \mathbf{e}_{i-\lceil \frac{k-1}{2} \rceil : i+\lfloor \frac{k-1}{2} \rfloor} + b), \quad (1)$$

where $\mathbf{w}$ and b are the parameters of the convolutional kernel, K is the kernel size, and $\mathbf{e}_{i-\lceil \frac{k-1}{2} \rceil : i+\lfloor \frac{k-1}{2} \rfloor}$ represents the concatenation of the embeddings from $i - \lceil \frac{k-1}{2} \rceil$-th character to $i + \lfloor \frac{k-1}{2} \rfloor$-th character. We use multiple kernels with different kernel sizes and concatenate the outputs of these convolutional kernels as the feature representation for each character. The output of CNN layer is

$[\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_N]$, where $\mathbf{h}_i \in \mathcal{R}^F$, and F is the number of kernels in the CNN network.

The third layer is CRF [9]. Given a sentence $\mathbf{x} = [c_1, c_2, ..., c_N]$ and a tag sequence $\mathbf{y} = [y_1, y_2, ..., y_N]$, the score of sentence $\mathbf{x}$ having tag sequence $\mathbf{y}$ is formulated as follows:

$$s(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{N} U_{i,y_i} + \sum_{i=1}^{N-1} A_{y_i, y_{i+1}}, \tag{2}$$

where U_{i,y_i} represents the unary score of assigning tag y_i to the i-th character, and $A_{y_i, y_{i+1}}$ represents the score of jumping from tag y_i to tag y_{i+1}. The unary score $\mathbf{U}_i \in \mathcal{R}^T$ is formulated as:

$$\mathbf{U}_i = \mathbf{W}_c \mathbf{h}_i + \mathbf{b}_c, \tag{3}$$

where $\mathbf{W}_c \in \mathcal{R}^{T \times F}$ and $\mathbf{b}_c \in \mathcal{R}^T$ are trainable parameters, and T is the size of the tag set. Then the likelihood probability of sentence $\mathbf{x}$ having tag sequence $\mathbf{y}$ is defined as:

$$p(\mathbf{y}|\mathbf{x}) = \frac{e^{\mathbf{s}(\mathbf{x},\mathbf{y})}}{\sum_{\mathbf{y}' \in \mathcal{Y}_\mathbf{x}} e^{\mathbf{s}(\mathbf{x},\mathbf{y}')}}, \tag{4}$$

where $\mathcal{Y}_\mathbf{x}$ represents the set of all possible tag sequences of sentence $\mathbf{x}$. And the loss function is formulated as follows:

$$\mathcal{L}(\boldsymbol{\theta}) = -\sum_{i=1}^{N_l} \log(p(\mathbf{y}_i|\mathbf{x}_i; \boldsymbol{\theta})), \tag{5}$$

where $\mathbf{x}_i$ is the i-th training sentence, $\mathbf{y}_i$ is its ground truth tag sequence, N_l is the number of labeled sentences in training set, and $\boldsymbol{\theta}$ represents all parameters of the neural CWS model.

3.2 Neural CWS with Lexicon and Unlabeled Data via Posterior Regularization

In this section we introduce our approach to exploit lexicon and unlabeled data to train a neural CWS model. Our approach is based on posterior regularization [6], and we propose a unified framework to incorporate the useful information in lexicon and unlabeled data as indirect supervision into model training by regularizing the prediction space of neural CWS model. In our approach the neural CWS model is trained in an iterative manner. Following [24], in iteration t, the loss function of the indirect supervision is:

$$\mathcal{L}^{PR}(\boldsymbol{\theta}) = \sum_{i=1}^{N_u} \mathrm{KL}(Q(\mathbf{y}|\mathbf{x}_i; \boldsymbol{D}, \boldsymbol{\theta}^t)||p(\mathbf{y}|\mathbf{x}_i; \boldsymbol{\theta})), \tag{6}$$

where $\boldsymbol{\theta}$ is the parameter set of CNN-CRF model, $\boldsymbol{\theta}^t$ is the model learned in iteration $t-1$, $\boldsymbol{D}$ represents Chinese lexicon, and N_u is the number of unlabeled sentences. KL is the KL divergence function. $Q(\mathbf{y}|\mathbf{x}; \boldsymbol{D}, \boldsymbol{\theta}^t)$ is the probability distribution of tag sequence $\mathbf{y}$ for unlabeled sentence $\mathbf{x}$ given lexicon $\boldsymbol{D}$ and previous model $\boldsymbol{\theta}^t$:

$$Q(\mathbf{y}|\mathbf{x}; \boldsymbol{D}, \boldsymbol{\theta}^t) = \frac{\exp(\phi(\mathbf{y}, \mathbf{x}; \boldsymbol{D}, \boldsymbol{\theta}^t))}{\sum_{\hat{\mathbf{y}} \in \mathcal{Y}(\mathbf{x})} \exp(\phi(\hat{\mathbf{y}}, \mathbf{x}; \boldsymbol{D}, \boldsymbol{\theta}^t))}, \tag{7}$$

where $\phi(\mathbf{y}, \mathbf{x}; \mathbf{D}, \boldsymbol{\theta}^t)$ is a score function of tag sequence $\mathbf{y}$ for sentence $\mathbf{x}$, and $\mathcal{Y}(\mathbf{x})$ is the set of all possible tag sequences of $\mathbf{x}$. $\phi(\mathbf{y}; \mathbf{x}, \mathbf{D}, \boldsymbol{\theta}^t)$ is designed to encode both lexicon information and the predictions of previous CWS model towards this sentence:

$$\phi(\mathbf{y}; \mathbf{x}, \boldsymbol{D}, \boldsymbol{\theta}^t) = \frac{n(\mathbf{x}, \mathbf{y}; \boldsymbol{D})}{n(\mathbf{x}, \mathbf{y})} + \alpha \cdot s(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta}^t), \tag{8}$$

where $n(\mathbf{x}, \mathbf{y})$ is the number of words in the segmentation result, $n(\mathbf{x}, \mathbf{y}; \boldsymbol{D})$ is the number of words in segmentation result which are included in lexicon $\boldsymbol{D}$, $s(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta}^t)$ is the segmentation score predicted by the CWS model $\boldsymbol{\theta}^t$ trained in previous iteration according to Eq. (2), and α is a positive coefficient. According to Eq. (8), if a tag sequence $\mathbf{y}$ for an unlabeled sentence $\mathbf{x}$ can lead to more lexicon-included words and has higher segmentation score according to existing CWS model, then it will have a higher probability in Eq. (7), and we regularize our neural CWS model so that it tends to generate this tag sequence in Eq. (6). In this way, the useful information in lexicon and unlabeled sentences can be incorporated into the learning of neural CWS model as indirect supervision.

Since a sentence usually has many possible tag sequences, following [24], KL function in Eq. (6) is approximated as:

$$\sum_{\mathbf{y} \in \mathcal{S}(\mathbf{x}_i)} \tilde{Q}(\mathbf{y}|\mathbf{x}_i; \boldsymbol{D}, \boldsymbol{\theta}^t) \log(\frac{\tilde{Q}(\mathbf{y}|\mathbf{x}_i; \boldsymbol{D}, \boldsymbol{\theta}^t)}{p(\mathbf{y}|\mathbf{x}_i; \boldsymbol{\theta})}), \tag{9}$$

where $\mathcal{S}(\mathbf{x}_i)$ is a subset of $\mathcal{Y}(\mathbf{x}_i)$ whose elements have the highest prediction scores according to existing CWS model $\boldsymbol{\theta}^t$. $\tilde{Q}$ is an approximation of Q on the subset $\mathcal{S}(\mathbf{x}_i)$ as follows:

$$\tilde{Q}(\mathbf{y}|\mathbf{x}_i; \boldsymbol{D}, \boldsymbol{\theta}^t) = \frac{\exp(\phi(\mathbf{y}, \mathbf{x}_i; \boldsymbol{D}, \boldsymbol{\theta}^t))}{\sum_{\hat{\mathbf{y}} \in \mathcal{S}(\mathbf{x}_i)} \exp(\phi(\hat{\mathbf{y}}, \mathbf{x}_i; \boldsymbol{D}, \boldsymbol{\theta}^t))}. \tag{10}$$

In our approach, in each iteration the neural CWS model $\boldsymbol{\theta}$ is updated based on both labeled sentences and the indirect supervision from lexicon and unlabeled data. The objective function for model update at iteration t is formulated as follows:

$$\mathcal{J}(\boldsymbol{\theta}) = -\sum_{i=1}^{N_l} \log(p(\mathbf{y}_i|\mathbf{x}_i; \boldsymbol{\theta})) - \lambda \sum_{i=1}^{N_u} \sum_{\hat{\mathbf{y}} \in \mathcal{S}(\mathbf{x}_i)} \tilde{Q}(\hat{\mathbf{y}}|\mathbf{x}_i; \boldsymbol{D}, \boldsymbol{\theta}^t) \log(p(\hat{\mathbf{y}}|\mathbf{x}_i; \boldsymbol{\theta})), \tag{11}$$

where λ is a positive coefficient to control the relative importance of indirect supervision in model training. In the first iteration, the initial neural CWS model $\boldsymbol{\theta}^1$ is trained on labeled sentences.

4 EXPERIMENT

4.1 Datasets and Experimental Settings

Two benchmark datasets released by the third international Chinese language processing bakeoff[1] [10] are used in our experiments. The first one is the *MSRA* dataset, which contains 46,364 labeled sentences for training and 4,365 labeled sentences for test. The second one is the *UPUC* dataset, which contains 18,804 and 5,117 labeled sentences for training and test.

The lexicon used in our experiments is the Sogou Chinese lexicon[2]. The size of character embeddings is 200. These character embeddings are pretrained on the Sogou news corpus[3] using the word2vec [16] tool. We use 400 convolutional kernels in the CNN network, and the sizes of these kernels vary from 2 to 5. λ in Eq. (11) and α in Eq. (8) are set to 0.5 and 1 respectively. We apply dropout technique to the embedding layer and the CNN layer, and the dropout rate is 0.3. RMSProp [4] algorithm is used for model training. The learning rate is 0.001 and the batch size is 64. These hyper-parameters are selected using validation data. Following [2],

[1] http://sighan.cs.uchicago.edu/bakeoff2006/download.html
[2] http://www.sogou.com/labs/resource/list_lan.php
[3] http://www.sogou.com/labs/resource/ca.php

we use the last 10% sentences in the training set as validation data, and the remaining labeled sentences for model training. In addition, we randomly sample 50% training data as unlabeled data. Each experiment is repeated 5 times and the average results are reported.

4.2 Performance Evaluation

In this section we evaluate the performance of our approach by comparing it with many baseline methods for Chinese word segmentation. These methods include: (1) LSTM-CRF, the most popular neural method for CWS based on the LSTM-CRF architecture [18]; (2) CNN-CRF, the neural CWS method based on the CNN-CRF architecture, which is the basic model in our approach; (3) Chen [2], a neural CWS method using LSTM to learn character features and also considering local contexts; (4) Zhang [27], an LSTM-CRF based CWS method which integrates lexicon into model training via feature templates; (5) Liu [12], a neural CWS method which incorporates lexicon into model training via multi-task learning; (6) Liu [13], a CRF based CWS method which utilizes lexicon and unlabeled data to build partially labeled data for model training; (7) Zhao [31], an LSTM based CWS method which incorporates lexicon and unlabeled data via building partially labeled data; (8) LUPR, our proposed neural CWS approach with both lexicon and unlabeled data via posterior regularization. We conducted experiments on different ratios of training data, and the experimental results of different methods are summarized in Tables 1 and 2.

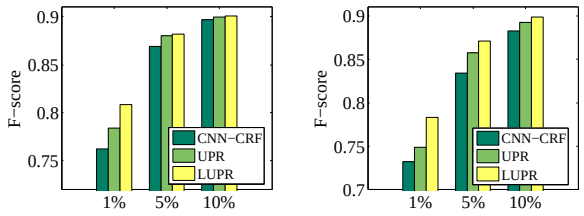
Table 1: The results on the *MSRA* dataset. P, R and F represent precision, recall and Fscore respectively.

	1%			5%			10%		
	P	*R*	*F*	*P*	*R*	*F*	*P*	*R*	*F*
LSTM-CRF	75.87	76.18	76.01	82.81	82.18	82.49	85.24	84.68	84.95
CNN-CRF	77.19	75.35	76.26	87.19	86.64	86.91	89.95	89.48	89.71
Chen [2]	77.20	74.32	75.73	84.19	83.44	83.80	87.50	86.05	86.76
Zhang [27]	76.64	76.55	76.60	87.15	86.73	86.94	89.49	89.10	89.29
Liu [12]	78.06	77.55	77.80	87.60	86.50	87.05	90.06	89.47	89.77
Liu [13]	81.48	78.92	80.18	83.76	81.58	82.66	85.20	83.09	84.13
Zhao [31]	80.68	**80.26**	80.47	86.94	85.67	86.30	88.69	87.21	87.94
LUPR	**81.98**	79.74	**80.84**	**88.42**	**87.92**	**88.17**	**90.35**	**89.83**	**90.09**

Table 2: The results on the *UPUC* dataset.

	1%			5%			10%		
	P	*R*	*F*	*P*	*R*	*F*	*P*	*R*	*F*
LSTM-CRF	70.49	73.44	71.92	79.66	80.95	80.29	82.97	84.73	83.84
CNN-CRF	72.03	74.50	73.22	82.60	84.23	83.40	87.75	88.79	88.27
Chen [2]	73.04	74.24	73.61	80.87	81.56	81.20	85.03	87.08	86.04
Zhang [27]	74.93	73.45	74.18	84.39	85.94	85.15	88.15	89.05	88.60
Liu [12]	73.01	76.33	74.63	83.82	85.70	84.75	87.71	89.27	88.48
Liu [13]	**79.50**	75.65	77.53	81.83	78.25	80.00	83.39	80.22	81.77
Zhao [31]	77.11	77.57	77.34	82.70	82.07	82.39	85.19	86.07	85.63
LUPR	78.18	**78.49**	**78.33**	**86.89**	**87.36**	**87.13**	**89.48**	**90.29**	**89.88**

According to Tables 1 and 2, our approach can outperform many neural Chinese word segmentation methods such as CNN-CRF, LSTM-CRF and Chen [2]. In addition, the advantage of our approach over these baseline methods becomes bigger when the number of training samples decreases. This is because these methods rely on a large number of labeled sentences to train neural CWS models, and cannot exploit the useful information in lexicon and unlabeled data.



(a) *MSRA* dataset. (b) *UPUC* dataset.

Figure 2: The performance of the basic CNN-CRF model and our approach with only unlabeled data (UPR) and with both lexicon and unlabeled data (LUPR).

When training data is insufficient, it is very difficult for these methods to train accurate and robust CWS models. Since our approach can exploit the useful information in both lexicon and unlabeled data, it can reduce the dependence on labeled sentences and achieve better performance than these baseline methods.

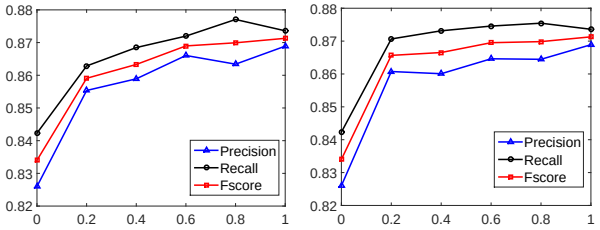
Although the methods proposed in [27] and [12] can also incorporate lexicon information into neural CWS model training, our approach can consistently outperform them. In [27] the lexicon is utilized via handcrafted feature templates, which need a lot of domain knowledge to design and may not be optimal. In addition, an extra LSTM network is incorporated to learn hidden character representations from these lexicon based features, making the neural model more difficult to train when labeled data is scarce. In [12] the lexicon is utilized via an auxiliary word classification task which is jointly trained with CWS model. Although these methods can utilize the lexicon for neural CWS, the useful information in massive unlabeled data is not considered. Our approach can exploit both lexicon and unlabeled data for neural Chinese word segmentation. Thus, our approach can consistently outperform them.

Although the methods in [13] and [31] can also exploit both lexicon and unlabeled data for CWS, our approach can still outperform them. In [13] and [31], the lexicon and unlabeled data are used to build partially annotated datasets. However, the partially annotated data constructed by word matching based on lexicons may contain heavy noise and the context information of sentences is not considered. In our approach, the lexicon and unlabeled data are used to provide indirect supervision for model training by regularizing the prediction space of the neural CWS model. The experimental results show that our approach is more effective in exploiting Chinese lexicon and unlabeled data for CWS than [13] and [31].

4.3 Effect of Lexicon and Unlabeled Data

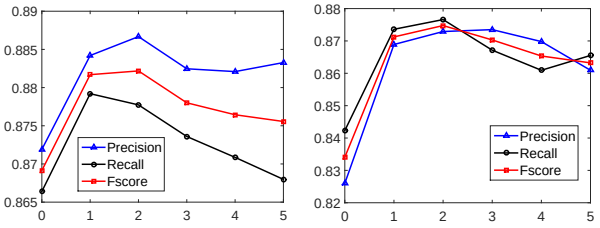
In this section, we conducted experiments to explore the effectiveness of lexicon and unlabeled data for neural Chinese word segmentation. The experimental results are summarized in Fig. 2.

According to Fig. 2, incorporating unlabeled data can effectively improve the performance of CWS in our approach. This is because the massive unlabeled data contains rich useful information for word segmentation. For example, if the character sequence "特朗普" frequently appear in unlabeled sentences with different contexts, then we can infer that it is probably a unique Chinese word. The result in Fig. 2 shows that our approach is effective in exploiting



(a) Ratio of unlabeled data. (b) Ratio of words in lexicon.

Figure 3: Influence of unlabeled data and lexicon size.



(a) *MSRA* dataset. (b) *UPUC* dataset.

Figure 4: Influence of S size in Eq. (9).

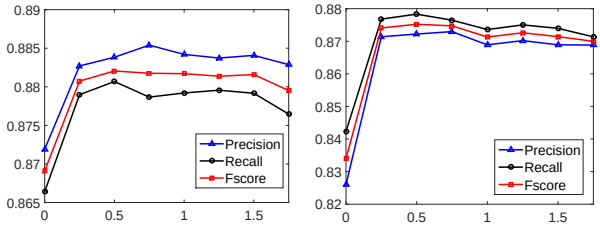
the useful information in unlabeled data for CWS. In addition, according to Fig. 2 after incorporating lexicon the performance of our approach can be further improved. This is because the Chinese lexicon can provide important information of whether a character sequence can be a valid Chinese word, which is useful for the CWS task. The result in Fig. 2 validates that our approach can effectively exploit Chinese lexicon to improve the performance of neural CWS.

4.4 Size of Lexicon and Unlabeled Data

In this section, we conduct experiments to explore the influence of the sizes of unlabeled data and Chinese lexicon on the performance of our approach. The experiments were conducted on the *UPUC* dataset, and we randomly sampled 5% of the training data for model training. The experiments on the *MSRA* dataset show similar patterns. The experimental results of unlabeled data size are summarized in Fig. 3(a). We can see that as more unlabeled data is incorporated, the performance of our approach consistently improves. This result further validates that the unlabeled data contains a lot of useful information for Chinese word segmentation, and our approach is effective in exploiting unlabeled data for neural CWS methods. The experimental results of lexicon size are shown in Fig. 3(b). According to Fig. 3(b), as more Chinese words are included in the lexicon, the performance of our approach improves. This is because with more Chinese words in the lexicon, our approach can have better capacity in recognizing the boundaries of words which rarely or never appear in training data but are included in the Chinese lexicon.

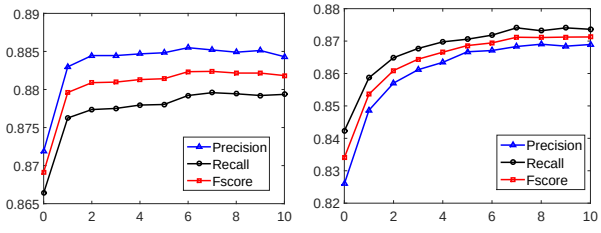
4.5 Influence of Hyper-parameters

In this section, we conducted experiments to explore the influence of three important hyper-parameters, i.e., the size of set S in Eq. (9), λ in



(a) *MSRA* dataset. (b) *UPUC* dataset.

Figure 5: Influence of λ in Eq. (11).



(a) *MSRA* dataset. (b) *UPUC* dataset.

Figure 6: Influence of iteration number.

Eq. (11) and iteration number, on the performance of our approach. The experimental results of the size of set S in Eq. (9) are shown in Fig. 4. In these experiments, we randomly sampled 5% of the training data for model training. According to Fig. 4, when the size of S in Eq. (9) increases from 0 to 1, the performance of our approach significantly improves. This is because when the size of S is 0, the useful information in unlabeled data and lexicon is not incorporated. Thus, the performance is not optimal. When the size of S becomes too large, the performance of our approach slightly decreases. It indicates that the number of informative tag sequences of unlabeled sentences inferred from lexicon is usually limited, and incorporating too many of them may introduce some noisy information. A moderate size of S (e.g., 2) is most suitable for our approach.

The experimental results of λ are shown in Fig. 5. λ is used to control the relative importance of the indirect supervision from unlabeled data and lexicon in Eq. (11). According to Fig. 5, as λ increases, the performance of our approach first increases and then slightly decreases. This is because when λ is too small, the useful information in the lexicon and unlabeled data is not fully exploited. Thus, the performance is not optimal. However, when λ becomes too large, the indirect supervision inferred from lexicon and unlabeled data is over-emphasized and the labeled sentences are not fully respected. Thus, the performance starts to decline. A moderate value of λ is most appropriate for our approach.

The experimental results of iteration number are shown in Fig. 6. According to Fig. 6, as the iteration number grows, the performance of our approach first improves and then gradually becomes stable. This is because in each iteration the neural CWS model can be enhanced by incorporating the unlabeled data and lexicon, and the refined CWS model in turn can help improve the indirect supervision in the next iteration of our approach. This result validates the

effectiveness of our approach in iteratively training neural CWS model by exploiting both labeled sentences and the indirect supervision inferred from the unlabeled sentences and Chinese lexicons.

4.6 Domain Adaptation for CWS

In Chinese word segmentation field, several domains (e.g., news) have accumulated much labeled data, while in many other domains (e.g., medical records) labeled data for CWS is scarce and even nonexistent. Although annotating sufficient labeled data for these domains is time-consuming and expensive, the unlabeled sentences are usually easy to collect. In addition, in many target domains there are off-the-shelf lexicons or it is relatively easy to build one. Thus, an interesting application of our approach is domain adaptation for Chinese word segmentation, i.e., using labeled sentences in a source domain (e.g., news) and unlabeled sentences in a target domain (e.g., medical records) as well as the lexicon from the target domain to train a robust neural CWS model for target domain. In this section we conduct experiments to explore the performance of our approach in domain adaptation for word segmentation.

Two datasets are used in our experiments. The first one is the *Zhuxian* dataset[4] built by Zhang et al. [25] from a Chinese online novel. We used the same lexicon[5] as [31] for this domain. In addition, we used the same unlabeled data as [13] which contains about 16K sentences. The second dataset is the Weibo dataset[6] released by the Weibo word segmentation task of NLPCC2016 [19]. This dataset contains 20,135 sentences for training and 8,592 sentences for test. Since there is no off-the-shelf lexicon for Weibo word segmentation, we built one using the words extracted from the training data. In addition, we regarded the training sentences as unlabeled data. We used *Zhuxian* and *Weibo* datasets as two target domains, and used the *UPUC* dataset as source domain. The experimental results are summarized in Table 3. The settings of our approach and baseline methods are consistent with previous experiments.

Table 3: Experimental results of domain adaptation.

	Zhuxian			Weibo		
	P	*R*	*F*	*P*	*R*	*F*
CNN-CRF	87.95	87.68	87.81	86.11	88.23	87.16
LSTM-CRF	85.94	86.26	86.10	86.39	88.35	87.36
Chen [2]	85.24	87.57	86.39	86.87	89.28	88.06
Zhang [27]	88.70	87.89	88.29	88.94	**90.89**	89.90
Liu [12]	87.79	88.04	87.91	86.24	88.49	87.35
Liu [13]	89.86	89.04	89.44	**90.31**	89.89	90.10
Zhao [31]	88.50	**91.32**	89.90	89.83	85.07	87.39
LUPR	**90.53**	89.39	**89.95**	89.85	90.73	**90.29**

According to Table 3, the performance of neural models that are trained on labeled data of source domain such as CNN-CRF, LSTM-CRF and [2] is relatively low in target domains. This is because there is huge difference in word distribution between source and target domains. Many words in target domain may not or rarely appear in the labeled data of source domain, making it difficult for these CWS methods to segment the sentences in target domain. The methods proposed in [27] and [12] which incorporate lexicon can effectively improve the performance of CWS in target domains. Our approach can outperform these methods because beyond the lexicon information our approach can also incorporate massive unlabeled data into neural model training, which can provide useful information for CWS. Although [13] and [31] can incorporate both lexicon and unlabeled data for word segmentation, our approach can outperform them in the domain adaptation scenario. This result implies that incorporating the lexicon and unlabeled data as indirect supervision via posterior regularization is more suitable for domain adaptation of CWS than utilizing them to build partially labeled data through word matching, which may introduce heavy noise.

[4]http://zhangmeishan. github.io/eacl14mszhang.zip
[5]This lexicon is crawled from http://baike.baidu.com/view/18277.htm
[6]https://github.com/FudanNLP/NLPCC-WordSeg-Weibo

4.7 Case Study

In this section, we conducted several case studies to further explore why our approach can improve the performance of CWS. We studied several cases in the domain adaptation experiments, where the source domain is *UPUC* and the target domain is *Weibo*. Several examples are illustrated in Table 4.

Table 4: Several examples of Chinese word segmentation.

	Example 1	Example 2
Sentence	养老金调整三大焦点问题	穆勒操刀梅开二度
CNN-CRF	养/老金/调整/三/大/焦点/问题	穆勒/操刀/梅开/二/度
LUPR	养老金/调整/三/大/焦点/问题	穆勒/操刀/梅开二度

According to Table 4, our approach performs better than baseline methods in domain adaptation scenario, especially on the sentences with OOV and rare words of source domain. For example, in the first example “养老金" is a rare word in the training data of source domain, and in the second example “梅开二度" is an OOV word in source domain. The segmentation results of the basic CNN-CRF model on these words are not correct. Since “养老金" is a popular word in the unlabeled data of target domain and “梅开二度" is included in the target domain lexicon, our approach can correctly segment these sentences. Thus, our approach can improve the performance of Chinese word segmentation by exploiting the useful information in both lexicon and unlabeled data.

5 CONCLUSION

In this paper, we propose a neural approach for Chinese word segmentation which can exploit the useful information in both Chinese lexicon and unlabeled sentences for model training. Our approach is based on the posterior regularization algorithm, and we propose a unified framework to incorporate both unlabeled data and lexicon to provide indirect supervision for model training by regularizing the prediction space of the neural CWS models. Extensive experiments are conducted on multiple benchmark datasets in both in-domain and cross-domain scenarios. The experimental results validate that our approach can effectively improve the performance of neural Chinese word segmentation.

ACKNOWLEDGMENTS

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFC1604002 and in part by the National Natural Science Foundation of China under Grant U1705261, Grant U1536201, Grant U1536207.

REFERENCES

[1] Deng Cai, Hai Zhao, Zhisong Zhang, Yuan Xin, Yongjian Wu, and Feiyue Huang. 2017. Fast and Accurate Neural Word Segmentation for Chinese. In *ACL*. 608–615.

[2] Xinchi Chen, Xipeng Qiu, Chenxi Zhu, Pengfei Liu, and Xuanjing Huang. 2015. Long short-term memory neural networks for chinese word segmentation. In *EMNLP*. 1197–1206.

[3] Xinchi Chen, Zhan Shi, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial Multi-Criteria Learning for Chinese Word Segmentation. In *ACL*. 1193–1203.

[4] Yann Dauphin, Harm de Vries, and Yoshua Bengio. 2015. Equilibrated adaptive learning rates for non-convex optimization. In *NIPS*. 1504–1512.

[5] Schubert Foo and Hui Li. 2004. Chinese word segmentation and its effect on information retrieval. *Information Processing & Management* 40, 1 (2004), 161–190.

[6] Kuzman Ganchev, Jennifer Gillenwater, Ben Taskar, et al. 2010. Posterior regularization for structured latent variable models. *Journal of Machine Learning Research* 11, Jul (2010), 2001–2049.

[7] Jianfeng Gao, Mu Li, Andi Wu, and Chang-Ning Huang. 2005. Chinese word segmentation and named entity recognition: A pragmatic approach. *Computational Linguistics* 31, 4 (2005), 531–574.

[8] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation* 9, 8 (1997), 1735–1780.

[9] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *ICML*. Morgan Kaufmann, 282–289.

[10] Gina-Anne Levow. 2006. The Third International Chinese Language Processing Bakeoff: Word Segmentation and Named Entity Recognition. In *SIGHAN@COLING/ACL*. Association for Computational Linguistics, 108–117.

[11] Zhongguo Li and Maosong Sun. 2009. Punctuation as implicit annotations for chinese word segmentation. *Computational Linguistics* 35, 4 (2009), 505–512.

[12] Junxin Liu, Fangzhao Wu, Chuhan Wu, Yongfeng Huang, and Xing Xie. 2018. Neural Chinese Word Segmentation with Dictionary Knowledge. In *NLPCC*. 80–91.

[13] Yijia Liu, Yue Zhang, Wanxiang Che, Ting Liu, and Fan Wu. 2014. Domain adaptation for CRF-based Chinese word segmentation using free annotations. In *EMNLP*. 864–874.

[14] Jin Kiat Low, Hwee Tou Ng, and Wenyuan Guo. 2005. A Maximum Entropy Approach to Chinese Word Segmentation. In *SIGHAN@IJCNLP 2005*. ACL, 161–164.

[15] Wencan Luo and Fan Yang. 2016. An Empirical Study of Automatic Chinese Word Segmentation for Spoken Language Understanding and Named Entity Recognition. In *NAACL*. 238–248.

[16] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *CoRR* abs/1301.3781 (2013).

[17] Wenzhe Pei, Tao Ge, and Baobao Chang. 2014. Max-margin tensor neural network for Chinese word segmentation. In *ACL*. 293–303.

[18] Nanyun Peng and Mark Dredze. 2017. Multi-task Domain Adaptation for Sequence Tagging. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*. 91–100.

[19] Xipeng Qiu, Peng Qian, and Zhan Shi. 2016. Overview of the NLPCC-ICCPOL 2016 Shared Task: Chinese Word Segmentation for Micro-Blog Texts. In *NLPCC/ICCPOL (Lecture Notes in Computer Science)*, Vol. 10102. Springer, 901–906.

[20] Weiwei Sun and Jia Xu. 2011. Enhancing Chinese word segmentation using unlabeled data. In *EMNLP*. 970–979.

[21] Jingjing Xu, Shuming Ma, Yi Zhang, Bingzhen Wei, Xiaoyan Cai, and Xu Sun. 2017. Transfer Deep Learning for Low-Resource Chinese Word Segmentation with a Novel Neural Network. In *NLPCC (Lecture Notes in Computer Science)*, Vol. 10619. Springer, 721–730.

[22] Nianwen Xue. 2003. Chinese word segmentation as character tagging. *IJCLCLP* 8, 1 (2003), 29–48.

[23] Jie Yang, Yue Zhang, and Fei Dong. 2017. Neural Word Segmentation with Rich Pretraining. In *ACL*. 839–849.

[24] Jiacheng Zhang, Yang Liu, Huanbo Luan, Jingfang Xu, and Maosong Sun. 2017. Prior knowledge integration for neural machine translation using posterior regularization. In *ACL*. 1514–1523.

[25] Meishan Zhang, Yue Zhang, Wanxiang Che, and Ting Liu. 2014. Type-supervised domain adaptation for joint segmentation and pos-tagging. In *EACL*. 588–597.

[26] Meishan Zhang, Yue Zhang, and Guohong Fu. 2016. Transition-based neural word segmentation. In *ACL*, Vol. 1. 421–431.

[27] Qi Zhang, Xiaoyu Liu, and Jinlan Fu. 2018. Neural Networks Incorporating Dictionaries for Chinese Word Segmentation. In *AAAI*. 5682–5689.

[28] Yanna Zhang, Jinan Xu, Guoyi Miao, Yufeng Chen, and Yujie Zhang. 2018. Addressing Domain Adaptation for Chinese Word Segmentation with Instances-Based Transfer Learning. In *CCL (Lecture Notes in Computer Science)*, Vol. 11221. Springer, 24–36.

[29] Hai Zhao, Changning Huang, and Mu Li. 2006. An Improved Chinese Word Segmentation System with Conditional Random Field. In *SIGHAN@COLING/ACL*. Association for Computational Linguistics, 162–165.

[30] Hai Zhao, Changning Huang, Mu Li, and Bao-Liang Lu. 2006. Effective Tag Set Selection in Chinese Word Segmentation via Conditional Random Field Modeling. In *PACLIC*. ACL, 87–94.

[31] Lujun Zhao, Qi Zhang, Peng Wang, and Xiaoyu Liu. 2018. Neural Networks Incorporating Unlabeled and Partially-labeled Data for Cross-domain Chinese Word Segmentation. In *IJCAI*. 4602–4608.

[32] Xiaoqing Zheng, Hanyang Chen, and Tianyu Xu. 2013. Deep learning for Chinese word segmentation and POS tagging. In *EMNLP*. 647–657.