Qun N, Yan H, Qiu XP *et al.* Chinese word segmentation via BiLSTM+Semi-CRF with relay node. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 35(5): 1115–1126 Sept. 2020. DOI 10.1007/s11390-020-9576-4

Chinese Word Segmentation via BiLSTM+Semi-CRF with Relay Node

Nuo Qun[1,2,#], Hang Yan[1,2,#], Xi-Peng Qiu[1,2,*], *Member*, *CCF*, and Xuan-Jing Huang[1,2], *Member*, *CCF*

[1] *School of Computer Science, Fudan University, Shanghai 200433, China*
[2] *Shanghai Key Laboratory of Intelligent Information Processing, Fudan University, Shanghai 200433, China*

E-mail: {14110240023, 11300720199, xpqiu, xjhuang}@fudan.edu.cn

Received March 23, 2019; revised July 4, 2019.

Abstract Semi-Markov conditional random fields (Semi-CRFs) have been successfully utilized in many segmentation problems, including Chinese word segmentation (CWS). The advantage of Semi-CRF lies in its inherent ability to exploit properties of segments instead of individual elements of sequences. Despite its theoretical advantage, Semi-CRF is still not the best choice for CWS because its computation complexity is quadratic to the sentence's length. In this paper, we propose a simple yet effective framework to help Semi-CRF achieve comparable performance with CRF-based models under similar computation complexity. Specifically, we first adopt a bi-directional long short-term memory (BiLSTM) on character level to model the context information, and then use simple but effective fusion layer to represent the segment information. Besides, to model arbitrarily long segments within linear time complexity, we also propose a new model named Semi-CRF-Relay. The direct modeling of segments makes the combination with word features easy and the CWS performance can be enhanced merely by adding publicly available pre-trained word embeddings. Experiments on four popular CWS datasets show the effectiveness of our proposed methods. The source codes and pre-trained embeddings of this paper are available on https://github.com/fastnlp/fastNLP/.

Keywords Semi-Markov conditional random field (Semi-CRF), Chinese word segmentation, bi-directional long short-term memory, deep learning

1 Introduction

The lack of obvious boundaries between Chinese words makes Chinese word segmentation (CWS) an important and preliminary pre-process step for Chinese natural language processing (NLP). Currently, a popular framework is considering CWS as a sequence labeling problem [1], and each character is assigned a segmentation tag to indicate its relative position inside the word. Therefore, conditional random field (CRF) [2] is widely used to predict the sequence of segmentation tags. Recently, various neural models [3–6] have introduced neural networks to learn features automatically, which alleviate the efforts in feature engineering. However, these character-based methods are difficult to utilize the word-level features. Some researchers make great efforts to incorporate word-level information for CWS [7–11]. Among them, Semi-Markov conditional random field (Semi-CRF) [12] is a very exciting model to find the best segmentation. The Semi-CRF directly scores the entire candidate segmentation and can fully utilize both the character-level and word-level information. [13] argues that any global feature functions used in CRF can be transferred to its counterpart in Semi-CRF. Therefore, Semi-CRF is strictly more expressive than CRF. Owing to great capabilities of Semi-CRF, it has been widely applied for sequence labeling tasks.

Despite theoretical advantage of Semi-CRF over CRF, Semi-CRF is still not the best choice for the CWS task due to the following two limitations.

The first is segment representation. The represen-

Regular Paper

This work is supported by the National Natural Science Foundation of China under Grant Nos. 61751201 and 61672162, and the Shanghai Municipal Science and Technology Major Project under Grant Nos. 2018SHZDZX01 and ZJLab.

[#] Contributed Equally to the Paper
[*] Corresponding Author

tation of a segment has a significant influence on the performance of Semi-CRF; therefore, how to design sophisticated features is a key to Semi-CRF. Usually, the segment-level features are more flexible but sparse than character-level features. To alleviate this, the CRF-type features [13] or latent variables [14] are incorporated into Semi-CRF to tackle the CWS problem. Some deep learning based methods [11, 15] have also been proposed to represent a segment with a dense vector. By introducing the extra pre-trained segment embeddings, the performance of Semi-CRF boosts remarkably. However, most of the existing models represent a segment by composing its containing characters and lose the rich segment-level contextual information.

The second is length limit. Semi-CRF needs to set a maximum length of a segment, L. When L is small, Semi-CRF cannot deal with the word with a length longer than L. However, a larger L does not guarantee the improvement of the performance of Semi-CRF and results in more time complexity. For CWS tasks, a large L is also a waste of computation resources since most of the Chinese words are composed of single character or two characters.

To address these two limitations, we propose a simple but effective architecture for CWS under the Semi-CRF framework. Our model consists of several key components: 1) a bi-directional long short-term memory (BiLSTM) [16] to capture the character-level contextual information; 2) an efficient fusion layer to extract the most valuable features of each segment; 3) a new decoding algorithm to enable Semi-CRF to deal with segments with arbitrary length. Experimental results on four popular datasets show the effectiveness of our proposed model.

Our contributions can be summarized as follows.

1) We propose a simple but effective BiLSTM+Semi-CRF architecture, which achieves comparable results with the CRF on four different datasets. Different with the previous models [11, 15], the BiLSTM encoder is used on the whole sequence rather than a single segment. Therefore, the representation of each segment is composed by not only its intra-characters, but also the contextual characters.

2) We propose a new decoding algorithm to allow Semi-CRF to handle arbitrarily long segment with linear computation costs. As far as we know, this is the first time that the Semi-CRF can handle arbitrarily long segment without increasing computation complexity.

3) The proposed Semi-CRF model can easily incorporate segment-level features into them. And the combination with publicly available word embeddings makes our model's $F1$ value outperform previous word-based models.

2 Background

Chinese word segmentation (CWS) is to segment a sequence of Chinese characters into a sequence of words. For the given sentence "姚明进入总决赛 (Yao Ming reaches the final)", it should be segmented as "姚明" (Yao Ming), "进入" (reaches), "总决赛" (the final). Currently, there are two popular approaches to solve the CWS task: character-based and word-based approaches.

2.1 Character-Based Approach

The character-based approach is to detect word boundaries which usually are represented by assigning labels to the characters in a sentence indicating their positions in a word. For example, the labels $\mathcal{L} = \{B, M, E, S\}$ are used to indicate whether a character is the beginning, middle, end of a word, or a word with a single character. In this setting, the sentence "姚明进入总决赛 (Yao Ming reaches the final)" can be labeled as "BEBEBME". Thus, the CWS task can be converted into a sequence labeling problem [1].

The character-based approach has been studied with considerable efforts in the NLP community. Since [1], this task is usually regarded as a sequence labeling problem. Recently, neural models [3–6, 17] have been widely employed for the CWS task for their ability to minimize the effort in feature engineering. These models utilize the more advanced neural models to extract features, such as LSTM [5], and gated recursive neural network (GRNN) [6].

The neural models are usually characterized by three specialized layers: 1) a character embedding layer; 2) an encoding layer, such as LSTM/CNN/GRNN; 3) an inference layer, such as CRF [2]. Fig.1 give an illustration of the popular BiLSTM+CRF architecture for CWS.

Although the character-based sequence labeling models have achieved great success, they cannot fully utilize the word-level information, such as word semantic representation and word length. To incorporate the word-level information, some previous work [18, 19] extends the character sequence into a directed acyclic graph (DAG) by adding "shortcut paths". A limitation

of these models is that the "shortcut paths" is added by matching a sentence with a large external lexicon. Therefore, their performances are easily affected by the quality of the external lexicon.

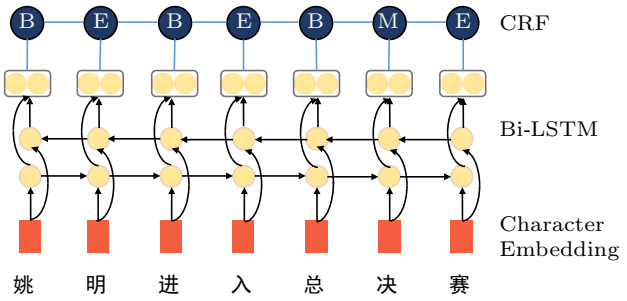


Fig.1. BiLSM+CRF architecture for CWS.

2.2 Word-Based Approach

The word-based approach is to directly model the word-level feature for CWS. There are two kinds of word-based CWS: 1) greedy transition based methods [7, 10] and 2) global segment-based methods [9, 11, 15]. In this paper, we focus on the segment-based methods.

In this paper, we focus on the global segment-based methods since it usually leads to better performance. For the above example, its output is [(1,2), (3, 4), (5, 7)], and every interval in this array represents the start and end of a word.

Formally, for a Chinese character sequence $x_{1:T} = x_1, x_2, \cdots, x_T$, its segmentation is defined as $s_{1:K} = s_1, s_2, \cdots, s_K$ where $s_k = (u_k, v_k)$ is a segment. u_k and v_k denote the start and the end position of s_k with regard to the original input sequence respectively. Besides, $1 \leqslant u_k \leqslant v_k \leqslant n$ and $u_{k+1} = v_k + 1$.

The conditional probability of a segmentation $\boldsymbol{s}_{1:K}$ can be formulated by Semi-CRF [12]. Here, we use the 0-order Semi-CRF, since there is no segmentation label in the CWS task.

$$p(s_{1:K}|x_{1:T}, \theta) = \frac{1}{Z(x_{1:T})} \exp\Big(F_\theta(x_{1:T}, s_{1:K})\Big) = \frac{1}{Z(x_{1:T})} \exp\Big(\sum_{k=1}^{K} f_\theta(x_{1:T}, s_k)\Big), \quad (1)$$

where $f_\theta(x_{1:T}, s_k)$ is a segment-level score function; θ denotes the parameters; $Z(x_{1:T})$ is a partition function which sums up scores for all possible segmentations.

$$Z(x_{1:T}) = \sum_{s_{1:K} \in \mathcal{S}} \exp(\sum_{k=1}^{K} f_\theta(x_{1:T}, s_k)),$$

where $\mathcal{S}$ is all possible segmentations. The partition function $Z(x_{1:T})$ can be also efficiently computed by the dynamic programming algorithm.

Although Semi-CRF is effective to utilize segmental-level information, previous Semi-CRF methods for CWS [11, 15] suffer from the following limitations.

1) Since we need to consider all the possible candidate segmentations, the computation complexity of Semi-CRF is $O(T^2)$ for a sentence with the length of T. Therefore, we usually set a maximum length L for the candidate segments. Although the time complexity can reduce to $O(LT)$, the segments with a length larger than L are discarded.

2) The segment score is usually calculated from inner features of a segment individually without considering the relation between segments.

3 Proposed Method

A key factor of Semi-CRF is how to construct the potential function $f_\theta(x_{1:T}, s_k)$ in (1). Generally, $f_\theta(x_{1:T}, s_k)$ can be computed by two types of features: 1) the character-level features; and 2) the word-level (or segment-level) features which represent information such as "the semantics of the word".

Recently, neural networks have been proven to be very favorable in CWS, and can significantly reduce the efforts of manual feature engineering. Existing methods [11, 15] use various neural models to represent segmental features. These segmental features are captured from the internal information within a segment and ignore contextual information of the segment.

Different from these existing methods, we propose a new model to effectively utilize the contextual and segmental-level features. The architecture of our method is illustrated in Fig.2, which can be divided into three layers: 1) a BiLSTM layer to extract the character-level contextual feature; 2) a fusion layer to extract the segment feature; and 3) a Semi-CRF layer for inference and decoding.

3.1 BiLSTM Layer

We first use bi-directional long short-term memory (BiLSTM) [16] on the input sequence of characters to model contextual information, which already becomes one of the most common choices for neural CWS encoding layers.

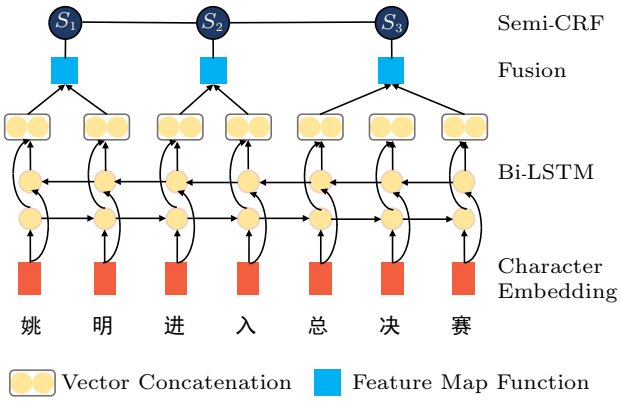


Fig.2. Proposed BiLSM+Semi-CRF architecture.

3.1.1 LSTM

LSTM is an extension of the recurrent neural network (RNN) [20], which aims to avoid the problems of gradient vanishing and explosion, and is very suitable to carry the long-term dependencies between characters.

Let $\boldsymbol{x}_t \in \mathbb{R}^{d_c}$ denote the embedding vector of character x_t. LSTM introduces memory cell $\boldsymbol{c} \in \mathbb{R}^{d_h}$ controlled by input gate $\boldsymbol{i} \in \mathbb{R}^{d_h}$, forget gate $\boldsymbol{f} \in \mathbb{R}^{d_h}$ and output gate $\boldsymbol{o} \in \mathbb{R}^{d_h}$. Thus, the hidden state $\boldsymbol{h}_t \in \mathbb{R}^{d_h}$ of the t-th character would be calculated as:

$$\begin{pmatrix} \boldsymbol{i}_t \\ \boldsymbol{o}_t \\ \boldsymbol{f}_t \\ \hat{\boldsymbol{c}}_t \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} \left(\boldsymbol{W}_g \begin{pmatrix} \boldsymbol{x}_t \\ \boldsymbol{h}_{t-1} \end{pmatrix} + \boldsymbol{b}_g \right),$$
$$\boldsymbol{c}_t = \boldsymbol{c}_{t-1} \odot \boldsymbol{f}_t + \hat{\boldsymbol{c}}_t \odot \boldsymbol{i}_t,$$
$$\boldsymbol{h}_t = \boldsymbol{o}_t \odot \tanh(\boldsymbol{c}_t),$$

where $\boldsymbol{W}_g \in \mathbb{R}^{4d_h \times (d_c + d_h)}$ and $\boldsymbol{b}_g \in \mathbb{R}^{4d_h}$ are trainable parameters, and $\sigma(\cdot)$ is sigmoid function.

3.1.2 BiLSTM

To utilize the rich contextual information, we employ the bi-directional LSTM (BiLSTM) neural network to encode the sequence of characters from both forward and backward. Specifically, each hidden state of BiLSTM is formalized as:

$$\boldsymbol{h}_i = \overrightarrow{\boldsymbol{h}}_i \oplus \overleftarrow{\boldsymbol{h}}_i,$$

where operator $\oplus$ indicates concatenation operation; and $\overrightarrow{\boldsymbol{h}}_t$ and $\overleftarrow{\boldsymbol{h}}_t$ are hidden states at step t of forward and backward LSTMs respectively.

The role of the BiLSTM layer is to capture the context information. Although this information is extracted in the character level, it is enough to incorporate the context information into the following segment representation.

3.2 Fusion Layer

A fusion layer is to extract the segment representation. For a segment $s_k = (u_k, v_k)$, we construct its representation based on the hidden states of the BiLSTM layer. Although it is possible to design some complicated neural models to encode the segment representation, we prefer a simple but effective model considering the efficiency of the model.

In order to be efficient, the segment representation $\boldsymbol{s}_k$ is a simple concatenation of the hidden state at the end position v_k and a boundary feature vector $\boldsymbol{\varphi}(s_k)$,

$$\boldsymbol{s}_k = \boldsymbol{h}_{u_k} \oplus \boldsymbol{\varphi}(s_k), \tag{2}$$

where $\varphi(s_k)$ has two forms $\varphi_{\max}(s_k)$ and $\varphi_{\text{diff}}(s_k)$:

$$\begin{aligned} \varphi_{\max}(s_k) &= \max(\boldsymbol{h}_{u_k}, \boldsymbol{h}_{v_k}), \\ \varphi_{\text{diff}}(s_k) &= \boldsymbol{h}_{v_k} - \boldsymbol{h}_{u_k}. \end{aligned} \tag{3}$$

$\varphi_{\max}(s_k)$ indicates the significant features chosen from the start and the end position of a segment with the max-pooling operation, and $\varphi_{\text{diff}}(s_k)$ captures the difference between both ends. As discussed in [13], the boundary information is essential for segmentation. The comparison between these two boundary feature functions will be discussed in Section 4.

Incorporating Segment Embedding. Previous work [10, 11] found the segment embedding was also an effective feature. Since words are one kind of segments, it is natural to combine this information into Semi-CRF. We concatenate the segment embedding with the representation in (2):

$$\boldsymbol{s}_k = \boldsymbol{h}_{u_k} \oplus \varphi(s_k) \oplus \boldsymbol{e}_{s_k}, \tag{4}$$

where $\boldsymbol{e}_{s_k}$ is the embedding of segment s_k. Since the number of candidate segments is very large, it is unfeasible to model the embeddings for all the segments. In this paper, we use the Tencent Chinese word embedding to judge whether a segment is a word [21], and this will be discussed in more detail in Section 4.

It is notable that the segment representation $\boldsymbol{s}_k$ also indirectly incorporates the information of its contextual segments since its constituent characters have the contextual information outside itself.

3.3 Semi-CRF Layer

Finally, a Semi-CRF layer is used to model the conditional probability of a segmentation $s_{1:k}$ over $x_{1:T}$.

3.3.1 Score Function

For a segmentation $s_{1:k}$, the representation $\boldsymbol{s}_k$ of each segment s_k captures both the intrinsic and the contextual semantic features, and its score function $f_\theta(x_{1:T}, s_k)$ in (1) is calculated by

$$f_\theta(x_{1:T}, s_k) = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{s}_k + b_{\mathrm{len}(s_k)}, \tag{5}$$

where $\boldsymbol{w}$ is the weight vector, and $b_{\mathrm{len}(s_k)}$ is the bias term and varies based on the length of s_k.

We can use the same idea as the forward algorithm used in the optimization of hidden Markov model to sum up all segmentation scores. And based on (1), the optimization goal is to maximize the log-likelihood over the training set. Given N training samples $\{x^{(n)}, s^{(n)}\}_{n=1}^{N}$, where $s^{(n)}$ is the gold segmentation of $x^{(n)}$, the objective function is defined as

$$\mathcal{L}(\theta) = \sum_{n=1}^{N} \log p(s^{(n)}|x^{(n)}, \theta),$$

where θ denotes all the trainable parameters.

3.3.2 Decoding

The decoding problem of Semi-CRF is to find the best segmentation of an input sequence $x_{1:T}$ with given parameters θ.

The predicted segmentation $\hat{s}$ has the highest probability among all the possible segmentations.

$$\begin{aligned}\hat{s} &= \operatorname*{argmax}_{s \in \mathcal{S}(x_{1:T})} p(s|x, \theta) \\ &= \operatorname*{argmax}_{s \in \mathcal{S}(x_{1:T})} \sum_{k=1}^{K} f_\theta(x, s_k),\end{aligned}$$

where $\mathcal{S}(x_{1:T})$ denotes the set of all possible segmentations of $x_{1:T}$.

Let L denote the maximum segment length and α_t denote the large segmentation score of the partial sequence $x_{1:t}$ among all the segmentations $\mathcal{S}(x_{1:t})$. α_t can be calculated by dynamic programming:

$$\alpha_t = \max_{l=1}^{L} \Big(\alpha_{t-l} + f_\theta(x_{1:T}, s_{(t-l+1):t})\Big), \tag{6}$$

where $s_{(t-l+1):t}$ denotes the segment of $x_{(t-l+1):t}$. Let $\alpha_0 = 0$ and $\alpha_{<0} = -\infty$. Thus, the segmentation corresponding to α_T is the best segmentation $\hat{s}$.

Although Semi-CRF is effective to utilize segmental-level information, it suffers from the quadratic computation complexity. If we do not set a upper bound for the segment length, calculating Semi-CRF will have a time complexity of $O(T^2)$, while an upper bound L can reduce it to $O(LT)$ (labels are not considered in the CWS case).

3.4 Semi-CRF Decoder with Relay Node

To alleviate the computation of large L, we further propose a new decoder, named Semi-CRF-Relay, which also has an upper bound L of segment length but is able to find segments with the length larger than L.

We define a special node, called relay node. If the current position is a relay node, its subsequent $L-1$ position cannot be the end of a segment. For example, if a segment is $x_t, x_{t+1}, \cdots, x_{t+L+1}$, its length is $L+2$ and larger than the upper bound L. To deal with this case, we let x_t and x_{t+1} be relay nodes, which means x_t and x_{t+1} must be merged into the segment $x_{(t+2):(t+L+1)}$.

Let α_t be the maximum score for a segmentation $s \in \mathcal{S}(x_{1:t})$ where the last segment ends at the t-th position. Let β_t be the maximum score of a segmentation where the t-th position is a relay node.

For simplicity, let $\gamma_{u:v} = f_\theta(x_{1:T}, s_{u:v})$ be the score function for segment $s_{u:v}$. α_t and β_t can be calculated recursively by

$$\alpha'_t = \max_{l=1}^{L} \Big(\alpha_{t-l} + \gamma_{(t-l+1):t}\Big), \tag{7}$$

$$\alpha_t = \max\Big(\alpha'_t, \beta_{t-L} + \gamma_{(t-L+1):t}\Big), \tag{8}$$

$$\beta_t = \max\Big(\alpha_{t-1} + f'(x_{u:v}, t), \beta_{t-1} + f'(x_{u:v}, t)\Big), \tag{9}$$

where $f'(x_{u:v}, t)$ is a score function indicating the current position t is a relay node. $f'(x_{u:v}, t)$ is calculated by

$$f'(x_{u:v}, t) = \boldsymbol{w}^{\mathrm{T}}(\boldsymbol{h}_t \oplus (\boldsymbol{h}_{t+L} - \boldsymbol{h}_t)) + b,$$

where $\boldsymbol{w}$ is weight vector and b is bias term. Here we set $\boldsymbol{w}$ and b to be different from (5).

By recursively computing (7)–(9), we can get the score of the best segmentation. Meanwhile, we can keep the segment information in P_α and P_β, which can been computed by

$$P_\alpha[t] = \operatorname*{argmax}_{l=1}^{L} \Big(\alpha_{t-l} + \gamma_{(t-l+1):t}\Big),$$

$$P_\beta[t-L] = I\Big(\alpha'_t < \beta_{t-L} + \gamma_{(t-L+1):t}\Big),$$

where $I(\cdot)$ is an indicator function, $P_\alpha[t] \in [1, L]$ denotes the length of the segment ending at position t, and $P_\beta[t] \in \{0, 1\}$ indicates whether the position t is a relay node.

The decoding algorithm is depicted in Algorithm 1 and Fig.3.

Algorithm 1. Decoding Algorithm for Semi-CRF-Relay

Input: sentence length: T; maximum segment length: L; P_α, P_β;
Function `Relay-Decoder`(P_α, P_β, L)**:**
  $t = T$;
  $segs = []$;
  while $t \geqslant 0$ **do**
    if $1 \leqslant P_\alpha[t] < L$ **then**
      $segs.append([t - P_\alpha[t], t])$;
      $t = t - P_\alpha[t]$;
    else
      $j = t - L$;
      while $j > 1 \&\& P_\beta[j-1] = 1$ **do**
        $j = j - 1$;
      end
      $segs.append([j, k])$;
      $t = j - 1$;
    end
  end
  return $segs[:: -1]$;

4 Experiment

4.1 Datasets

We evaluate our models on four popular CWS datasets from SIGHAN 2005 [22]. Basic dataset statistics are displayed in Table 1. The number of sentences is different from [22] because we randomly picked 10% samples as the development set. All continuous numbers and letters are replaced by special tags. The datasets PKU and MSRA are in simplified Chinese and CITYU and AS are in traditional Chinese. We map CITYU and AS into simplified Chinese.

As shown in Table 1, the MSRA dataset has a significantly higher ratio of long words. It is because that in MSRA name entities are viewed as one of the taxonomies of Chinese words. There exist very long name entities in this dataset. Therefore, the MSRA dataset is very challenging for Semi-CRF based CWS methods.

4.2 Experimental Configuration

We adopt the same hyper-parameters for all of our architectures and datasets. Adagrad [23] with learning rate of 0.037 is used. The models resulting in the high-

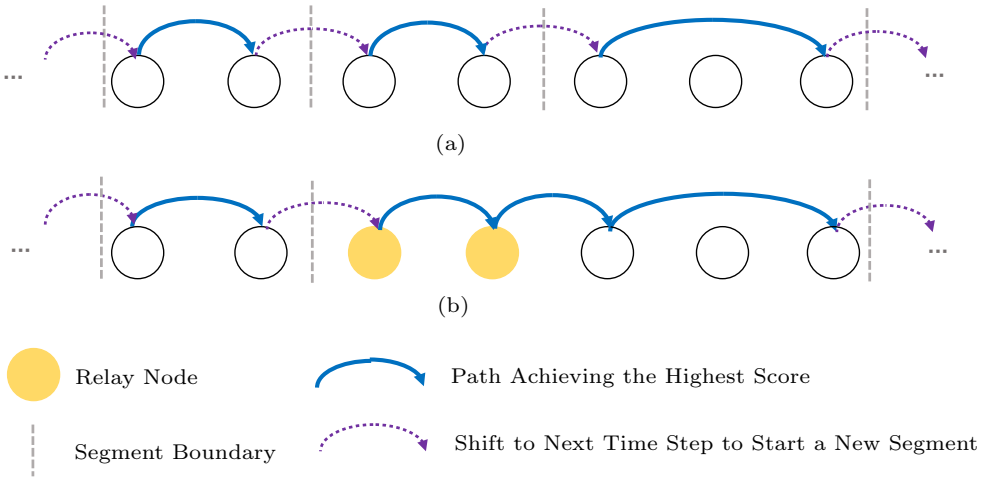


Fig.3. Decoding process for (a) Semi-CRF and (b) Semi-CRF-Relay.

Table 1. Details of Four Datasets in SIGHAN 2005

Dataset		Sent# ($\times 10^3$)	CharType# ($\times 10^3$)	BigramType# ($\times 10^3$)	R_{OOV} (%)	$R_{>4}$ (%)	$R_{>5}$ (%)	$R_{>6}$ (%)
PKU	Train	19.6	4.7	196.9	–	0.30	0.09	0.05
	Test	1.9	–	–	3.33	0.20	0.07	0.03
MSRA	Train	78.4	5.2	343.4	–	1.25	0.76	0.48
	Test	4.0	–	–	2.24	1.05	0.69	0.42
AS	Train	638.1	5.9	558.5	–	0.15	0.04	0.02
	Test	14.4	–	–	3.92	0.28	0.14	0.06
CITYU	Train	47.8	4.8	263.4	–	0.22	0.08	0.04
	Test	1.5	–	–	6.32	0.32	0.13	0.07

Note: Sent#, CharType# and BigramType# represent the numbers of sentences, unique characters and unique Bigrams respectively. R_{OOV} is the out-of-vocabulary ratio in test set. $R_{>4}$, $R_{>5}$, $R_{>6}$ are the ratio of words longer than 4, 5, 6 respectively.

est $F1$ score in the development set are used for evaluation. All embeddings are subject to a dropout layer with drop probability of 0.3 [24]. Parameters are initialized by Xavier uniform initializer [25]. Every experiment has been repeated at least five times, and then average performance was reported.

The rest hyper-parameters are shown in Table 2.

Table 2. Hyper-Parameter Settings

Hyper-Parameter	Value
Character/bigram embedding dimension d_c	100
Word embedding dimension d_w	200
BiLSTM hidden size d_h	200
Gradients clip	5
Batch size	32

Character Embeddings. As previous work [4, 5] reported, the bigram embeddings could significantly boost the performance of CWS. Following their work, we also represent each character x_i by concatenating its character embedding and adjacent bigram embeddings $\boldsymbol{x}_t = \boldsymbol{e}_{x_t} \oplus \boldsymbol{e}_{(x_{t-1},x_t)} \oplus \boldsymbol{e}_{(x_t,x_{t+1})}$.

The pre-trained character embeddings are extensively used in CWS tasks [4, 5, 11]. Besides the character embedding, we also pre-trained bigram embeddings. We pre-train unigram and bigram embeddings in Chinese Wikipedia corpus by [26] which improves standard word2vec by incorporating token order information. For a sentence with characters "abcd...", the character sequence is "a b c ..."; the bigram sequence is "ab bc cd ...". To quantitatively study the effect of pre-trained embeddings, we present experimental results in the four datasets with and without pre-trained embeddings in Table 3. Results show that pre-trained embeddings have positive effect in the performance. Therefore, we use pre-trained character and bigram embeddings through all experiments, and both embeddings are fine-tuned during the training process.

Segment Embeddings. Segment embeddings or external dictionaries are two widely used external data for word-based features [10, 27]. And [11] argues that pre-trained segment embeddings greatly benefit the performance of Semi-CRF. We adopt Tencent pre-trained word embeddings [21], whose entries are not guaranteed to be a legal word. The model has to learn how to exploit this kind of information. For example, "北京" has a vector, "到北京" also has a vector, although "到北京" is not word. If a segment has its corresponding entry in the pre-trained embedding, then $\boldsymbol{e}_{s_k}$ will be assigned as that vector. If a segment is not in the embedding, then a shared randomly initialized vector will be assigned to $\boldsymbol{e}_{s_k}$. In the Semi-CRF-Relay-Word scenario, if there exists at least one word longer than L in the embedding, a shared randomly initialized vector will be assigned to $\boldsymbol{e}_{s_k}$ when calculating relay scores; otherwise it is the same vector as a segment not in the embedding. Word embeddings are fixed for all the experiments.

Maximum Segment Length L. For the Semi-CRF models, the maximum segment length of L is a very important hyper-parameter. To overcome Semi-CRF's inability to cover long segments, we try to increase L. The variation curve between the $F1$ score and the maximum length of L for Semi-CRF is shown in Fig.4 in dotted line. Results are averaged from five experiments. The $F1$ score increases with the increment of L, and it plateaus between L in [10, 12]. While the Semi-CRF-Relay can achieve better results even with small L, which is depicted in solid line in Fig.4. Based on this observation and the dataset statistics, we use $L = 4$ for all of the Semi-CRF-Relay experiments, and $L = 6$ for all of the Semi-CRF experiments. Since the Semi-CRF cannot deal with the segment with a length larger than L, we drop the samples with a word longer than L in the training phase of BiLSTM+Semi-CRF model.

Boundary Feature Function. As discussed in Subsection 3.2, we test two kinds of feature functions. And the results are presented in Table 4. Although the preponderance of $\varphi_{\text{diff}}(s_k)$ is subtle, it outperforms $\varphi_{\max}(s_k)$ in all scenarios. Therefore we use $\varphi_{\text{diff}}(s_k)$ in all experiments.

Table 3. $F1$ Score with or Without Pre-Trained Character and Bigram Embeddings

	PKU	MSR	AS	CITYU
CRF	94.90(−0.02)	96.89(+**0.03**)	95.15(+**0.21**)	95.33(+**0.02**)
Semi-CRF	94.72(+**0.31**)	96.08(+**0.21**)	95.12(+**0.40**)	95.21(+**0.44**)
Semi-CRF-Relay	94.80(+**0.19**)	97.00(+**0.01**)	95.46(+**0.12**)	95.38(+**0.26**)

Note: Values are averaged from five experiments, and the number in the brackets indicates the improvement made by introducing pre-trained embeddings.

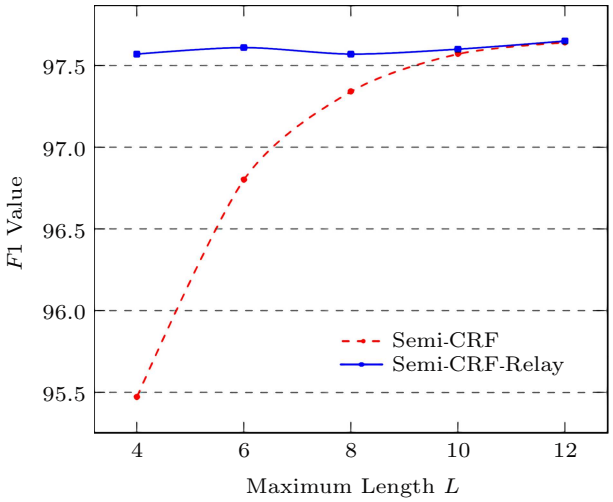


Fig.4. Average $F1$ score for different Ls in the development set of the MSR dataset.

4.3 Main Results

We compare our models with several previous neural CWS models, which can be categorized into the following two classes.

- *Character-Based Models.* This kind of models mainly uses various neural models to encode the feature for each character, and then assign a label with an MLP or CRF classifier. The compared models include max-margin tensor neural network (MMTNN) [4], gated recursive neural network (GRNN) [6], LSTM [5], and our implemented BiLSTM+CRF model.
- *Word-Based Models.* This kind of models mainly directly assigns scores (or probabilities) to different segmentations, in which various neural models are used to calculate the score of each segmentation. The compared models include segmental RNN (SRNN) [15], neural Semi-CRF [11], and neural segmentation scoring model (NSSM) [9].

By comparing the above models, we verify the effectiveness of our proposed model. Since some of the above models use the character-level features only, we also investigate the effectiveness of our models with both character-level features and additional word-level features, and their segment representations are (3) and (4) respectively.

The main results ($F1$ scores) are shown in Table 5. We first discuss the models in each sub-block, and then give an overall analysis.

Upper Block. The upper block lists the performance

Table 4. Comparison of Different Feature Functions

Model	PKU	MSRA	AS	CITYU	Average
Semi-CRF (max)	95.00(0.16)	96.19(0.19)	95.41(0.17)	95.55(0.18)	95.54
Semi-CRF-Relay (max)	94.82(0.13)	96.95(0.14)	95.54(0.09)	95.60(0.09)	95.73
Semi-CRF (diff)	**95.03(0.07)**	**96.29(0.13)**	**95.52(0.02)**	**95.65(0.05)**	**95.62**
Semi-CRF-Relay (diff)	**94.99(0.11)**	**97.01(0.06)**	**95.58(0.04)**	**95.65(0.07)**	**95.81**

Note: "max" and "diff" indicate two different methods to represent the segment embedding. Numbers in the brackets are standard deviation of five experiments. Bold results indicate that better performance than its counterpart.

Table 5. Comparisons Between BiLSTM+CRF, BiLSTM+Semi-CRF-Relay and Previous Models

Model	PKU	MSRA	AS	CITYU	Average
MMTNN [4]	95.2	97.2	-	-	-
GRNN [6]	94.5*	95.4	-	-	-
LSTM [5]	94.8*	95.6	-	-	-
BiLSTM+CRF (our implementation)	94.88(0.13)	96.92(0.07)	95.36(0.10)	95.35(0.04)	95.63
Neural Semi-CRF§ [11]	93.91	95.21	-	-	-
NSSM [9]	95.5	96.5	-	-	-
SRNN [15]	91.3	90.7	93.7	93.5	92.3
BiLSTM+Semi-CRF§	95.03(0.07)	96.29(0.13)	95.52(0.02)	95.65(0.05)	95.62
BiLSTM+Semi-CRF-Relay§	94.99(0.11)	97.01(0.06)	95.58(0.04)	95.65(0.07)	95.81
BiLSTM+Semi-CRF+SE	95.69(0.01)	96.99(0.01)	**95.90(0.04)**	96.22(0.04)	96.20
BiLSTM+Semi-CRF-Relay+SE	**95.75(0.06)**	**97.54(0.01)**	95.87(0.03)	**96.27(0.05)**	**96.36**

Note: The first four rows are the character-based models. The fifth row to the eighth row are the previous word-based models. The rest are our models. The number in the brackets is the standard deviation of five repeated experiments. We set the maximum length limit $L = 4$ for all of the Semi-CRF-Relay experiments, and $L = 6$ for all of the Semi-CRF experiments. § indicates models without the segment embedding $\boldsymbol{e}_{s_k}$. $*$: for a fair comparison, we use the results reported in [9], in which the preprocessing phase in the original work is not adopted. SE means "segment embeddings".

of the state-of-the-art character-based models.

Middle Block. The middle block lists the performance of the word-based models without utilizing word embeddings. These models use the character embeddings only. The segment representation is composed of its constituent characters without explicitly modeling the segment embedding.

Without utilizing word embeddings, the previous word-based models give relatively lower performances than the competitor character-based models, such as the BiLSMT+CRF. However, our proposed BiLSTM+Semi-CRF is comparable with BiLSTM+CRF on average and worse than BiLSTM+CRF on the MSR dataset. The behind reason is that the MSRA dataset has a high ratio of long words, and Semi-CRF cannot deal with words with length larger than $L = 6$. Increasing the maximum segment length L does not improve the performance for datasets without long words, but results in higher computation.

By introducing the relay node, BiLSTM+Semi-CRF-Relay outperforms its BiLSTM+Semi-CRF counterpart in the average $F1$ score. The boost mainly comes from the MSR dataset, and the coverage of arbitrarily long segments of Semi-CRF-Relay improves its $F1$ score notably, from 96.29 to 97.01.

Besides, BiLSTM+Semi-CRF also outperforms neural Semi-CRF model [11] even without complex segmental feature functions, which shows the effectiveness of the contextual segment representation in our architecture.

Bottom Block. The last two rows of Table 5 gives the $F1$ scores for our proposed models with extra word embeddings. As it shows, word embeddings are also beneficial to word-based models. The average $F1$ scores of our models are significantly improved by utilizing the extra word embeddings. Not like [11], the word embeddings used in our case are publicly available and do not need a trained baseline model to do CWS in unlabeled data first.

Overall Analysis. The incorporation of word-based features into CWS has been widely studied. But generally, although the word-based models can utilize the word-level features, the previous word-based models have not shown their superiority to the character-based models. For example, the neural Semi-CRF [11] without the pre-trained word embeddings just has a lower performance than most of the character-based models. But our proposed BiLSTM+Semi-CRF achieves comparable results with the character-based models. By introducing the relay node, our model can further improve its performance. Besides, the pre-trained word embeddings can be effortlessly added to our proposed Semi-CRF models to further enhance the performance.

4.4 Performances on OOV and Long Words

A main drawback of Semi-CRF is its inability to deal with arbitrarily long segments in linear time. Therefore, we propose Semi-CRF-Relay to tackle this issue. In this part, we would like to investigate the performance on the recall of long words and out-of-vocabulary (OOV) words of the BiLSTM+Semi-CRF-Relay when L is restricted to a small value.

As shown in Table 6, there are two noticeable facts. Firstly, even the maximum length limit L of Semi-CRF-Relay is 4, which means it can only model words with a

Table 6. Comparisons Between BiLSTM+CRF and BiLSTM+Semi-CRF-Relay in Recall for Long Words and Out-of-Vocabulary Words

Model		PKU	MSRA	AS	CITYU	Average
BiLSTM+CRF	F1	94.88	96.92	95.36	95.35	95.63
	OOV	56.69	62.10	56.73	66.40	60.48
	$R_{>4}$	78.47	80.88	56.10	66.15	70.40
	$R_{>5}$	78.07	78.90	41.27	64.20	65.61
	$R_{>6}$	79.63	75.43	23.19	66.67	61.23
BiLSTM+Semi-CRF-Relay*	F1	94.99(+0.11)	97.01(+0.10)	95.58(+0.22)	95.65(+0.30)	95.81(+0.18)
	OOV	59.82(+3.13)	60.88(−1.23)	62.09(+5.37)	69.55(+3.15)	63.08(+2.61)
	$R_{>4}$	78.15(−0.32)	81.26(+0.38)	56.77(+0.67)	67.50(+1.35)	70.92(+0.52)
	$R_{>5}$	79.39(+1.32)	79.58(+0.67)	42.66(+1.39)	62.75(−1.44)	66.09(+0.48)
	$R_{>6}$	79.63(−0.00)	76.84(+1.41)	28.02(+4.83)	65.56(−1.11)	62.51(+1.28)

Note: OOV means out-of-vocabulary recall. $R_{>4}$, $R_{>5}$, $R_{>6}$ are recalls for words longer than 4, 5, 6 respectively. Numbers in the brackets are improvement relative to the BiLSTM+CRF model. $*$: the maximum segment length limit $L = 4$.

length smaller than 5 without the help of the relay node. However, the recalls ($R_{>4}$, $R_{>5}$, $R_{>6}$) show that Semi-CRF-Relay has the ability to recognize long segments and overcomes the limitation of the segment length. Compared with CRF, Semi-CRF-Relay has better performance on OOV and long words.

Secondly, considering the $F1$ score and OOV recall, Semi-CRF-Relay is better than corresponding CRF models in most datasets. We assume the higher OOV recall is a sign that Semi-CRF-Relay learns a better representation of words.

To illustrate the effectiveness of Semi-CRF-Relay to handle the segment with an arbitrary length, we take seven example sentences from the test set of the MSR dataset. The model used to predict these sentences is BiLSTM+Semi-CRF-Relay with $L = 4$. The segmentation results of these sentences are shown in Fig.5. Different segments are separated by "·", and the underlined segments are longer than 4, which demonstrates Semi-CRF-Relay's ability to segment words longer than L. Especially, in the 7th sentence, Semi-CRF-Relay can even handle the long segment with length of 23.

5 Related Work

Semi-CRFs have been widely used in NER [28, 29] and Chinese new word identification [30].

There are two concerns about applying the Semi-CRF to solve the CWS task: the segment representation and restriction of segment length.

1) For the segment representation, various neural models, such as RNN [11, 15] and CNN [11], are utilized to encode the segment features. The segment representation is composed from its contained characters, ignoring the segment-level contextual information.

Different from the above models, we propose a more simple but effective architecture to encode the segment representation. We first use BiLSTM to capture the character-level contextual information, and then use a fusion layer to extract the segment representation. Thus, each segment can indirectly contain the information from its adjacent segments.

2) For the length limit, [28] combines the word phrase into segment units, in this way the word phrase can be in any length; however, the number of segment units in one segment is still restricted. [30] uses the NBest method to filter out a lot of segmentation before Semi-CRF finds the best one.

Different from the above methods, we design a new decoding algorithm to handle segments longer than the maximum length.

6 Conclusions

In this paper, we revisited the utilization of Semi-CRF in CWS tasks and proposed a simple but effective architecture BiLSTM+Semi-CRF, which obtains comparable results with CRF. To alleviate the performance decay caused by long segments, we introduced a new model named Semi-CRF-Relay, which is the first Semi-CRF based model that can be applied to arbitrarily long segments without causing quadratic computation complexity. The contrasts between the BiLSTM+CRF model and the BiLSTM+Semi-CRF-Relay model showed the latter slightly improves performance for all datasets and display similar performance in recalls for long words. Besides, the inherent ability of Semi-CRF to model segments makes word embeddings can be easily integrated into CWS models. Experiments revealed that this simple combination is quite fruitful.

现·为·南京大学社会学系·讲师·，·在职·博士生·。
延安市·是·典型·的·黄土高原·丘陵·沟壑·区·，·水土流失·面积·占·百分之七十八点四·，
设·在·江·边·的·葛洲坝股份公司截流总指挥部·，·是·一·间·临时·搭·起·的·活动·平房·。
中共中央政策研究室·副·主任·郑科扬·、·国务院三峡委办公室·副·主任·李世忠·也·随同·考察·。
中国人民银行上海市分行·、·深圳经济特区分行·要·组织·专门·人员·检查·、·监督·上海·、· 深圳证券交易所·及其·下属·证券·登记·结算·公司·、·有关·商业银行·的·证券·交易·清算·行为·。
5个·下属·公司·：·东大阿尔派软件股份有限公司·，·东东系统集成有限公司·，· 北国数据通信有限公司·，·东软经济技术发展有限公司·和·东龙经济技术服务有限公司·。
华中理工大学电子与信息工程系光电信息处理研究室·主任·张肇群·教授·完成·了·国家· <NUM> ·计 划· <NUM> ·主题·项目· "·光电·实时·目标·识别·与·跟踪·" ·课题·。

Note: Different segments are separated by "·". Examples are taken from the MSRA dataset, and the model is BiLSTM+Semi-CRF-Relay with the maximum segment length limit $L = 4$.

Fig.5. Case study of BiLSTM+Semi-CRF-Relay.

References

[1] Xue N. Chinese word segmentation as character tagging. *International Journal of Computational Linguistics and Chinese Language Processing*, 2003, 8(1): 29-48.

[2] Lafferty J D, McCallum A, Pereira F C N. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. the 18th International Conference on Machine Learning*, June 2001, pp.282-289.

[3] Zheng X, Chen H, Xu T. Deep learning for Chinese word segmentation and POS tagging. In *Proc. the 2013 Conference on Empirical Methods in Natural Language Processing*, October 2013, pp.647-657.

[4] Pei W, Ge T, Chang B. Max-margin tensor neural network for Chinese word segmentation. In *Proc. the 52nd Annual Meeting of the Association for Computational Linguistics*, June 2014, pp.293-303.

[5] Chen X, Qiu X, Zhu C, Liu P, Huang X. Long short-term memory neural networks for Chinese word segmentation. In *Proc. the 2015 Conference on Empirical Methods in Natural Language Processing*, September 2015, pp.1197-1206.

[6] Chen X, Qiu X, Zhu C, Huang X. Gated recursive neural network for Chinese word segmentation. In *Proc. the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, July 2015, pp.1744-1753.

[7] Zhang Y, Clark S. Chinese segmentation with a word-based perceptron algorithm. In *Proc. the 45th Annual Meeting of the Association for Computational Linguistics*, June 2007, pp.840-847.

[8] Sun W. Word-based and character-based word segmentation models: Comparison and combination. In *Proc. the 23rd International Conference on Computational Linguistics*, August 2010, pp.1211-1219.

[9] Cai D, Zhao H. Neural word segmentation learning for Chinese. In *Proc. the 54th Annual Meeting of the Association for Computational Linguistics*, August 2016, pp.409-420.

[10] Zhang M, Zhang Y, Fu G. Transition-based neural word segmentation. In *Proc. the 54th Annual Meeting of the Association for Computational Linguistics*, August 2016, pp.421-431.

[11] Liu Y, Che W, Guo J, Qin B, Liu T. Exploring segment representations for neural segmentation models. In *Proc. the 25th International Joint Conference on Artificial Intelligence*, July 2016, pp.2880-2886.

[12] Sarawagi S, Cohen W. Semi-Markov conditional random fields for information extraction. In *Proc. the Annual Conference on Neural Information Processing Systems*, December 2005, pp.1185-1192.

[13] Andrew G. A hybrid Markov/semi-Markov conditional random field for sequence segmentation. In *Proc. the 2006 Conference on Empirical Methods in Natural Language Processing*, July 2006, pp.465-472.

[14] Sun X, Zhang Y, Matsuzaki T, Tsuruoka Y, Tsujii J. A discriminative latent variable Chinese segmenter with hybrid word/character information. In *Proc. the 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, May 2009, pp.56-64.

[15] Kong L, Dyer C, Smith N A. Segmental recurrent neural networks. In *Proc. the 4th International Conference on Learning Representations*, May 2015.

[16] Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Computation*, 1997, 9(8): 1735-1780.

[17] Chen X, Shi Z, Qiu X, Huang X. Adversarial multi-criteria learning for Chinese word segmentation. In *Proc. the 55th Annual Meeting of the Association for Computational Linguistics*, July 2017, pp.1193-1203.

[18] Chen X, Shi Z, Qiu X, Huang X. DAG-based long short-term memory for neural word segmentation. arXiv:1707.00248, 2017. https://arxiv.org/abs/1707.00248, August 2019.

[19] Yang J, Zhang Y, Liang S. Subword encoding in Lattice LSTM for Chinese word segmentation. arXiv:1810.12594, 2018. https://arxiv.org/abs/1810.12594, August 2019.

[20] Elman J L. Finding structure in time. *Cognitive Science*, 1990, 14(2): 179-211.

[21] Song Y, Shi S, Li J, Zhang H. Directional skip-gram: Explicitly distinguishing left and right context for word embeddings. In *Proc. the 2018 Conference of the North American Chapter of the Association for Computational Linguistics*, June 2018, pp.175-180.

[22] Emerson T. The second international Chinese word segmentation bakeoff. In *Proc. the 4th SIGHAN Workshop on Chinese Language Processing*, June 2005, pp.123-133.

[23] Zeiler M D. ADADELTA: An adaptive learning rate method. arXiv:1212.5701, 2012. https://arxiv.org/abs/1212.5701, August 2019.

[24] Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 2014, 15(1): 1929-1958.

[25] Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks. In *Proc. the 13th International Conference on Artificial Intelligence and Statistics*, May 2010, pp.249-256.

[26] Ling W, Dyer C, Black A W, Trancoso I. Two/too simple adaptations of word2vec for syntax problems. In *Proc. the 2015 Conference of the North American Chapter of the Association for Computational Linguistics*, May 2015, pp.1299-1304.

[27] Zhang Q, Liu X, Fu J. Neural networks incorporating dictionaries for Chinese word segmentation. In *Proc. the 32nd AAAI Conference on Artificial Intelligence*, February 2018, pp.5682-5689.

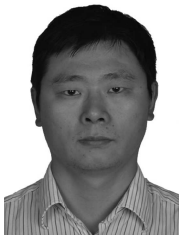
[28] Finkel J R, Manning C D. Nested named entity recognition. In *Proc. the 2009 Conference on Empirical Methods in Natural Language Processing*, August 2009, pp.141-150.

[29] Ye Z, Ling Z. Hybrid semi-Markov CRF for neural sequence labeling. In *Proc. the 56th Annual Meeting of the Association for Computational Linguistics*, July 2018, pp.235-240.

[30] Sun X, Huang D, Song H, Ren F. Chinese new word identification: A latent discriminative model with global features. *Journal of Computer Science and Technology*, 2011, 26(1): 14-24.

Nuo Qun received his M.S. degree in computer science from Handong Global University, Puhang, Korea, in 2004. Currently he is a Ph.D. candidate in School of Computer Science, Fudan University, Shanghai. He is also an associated professor in School of Information Science and Technology, Tibet University, Lhasa. His research interests include natural language processing and deep learning.

Hang Yan received his B.S. degree in information science from Fudan University, Shanghai, in 2015. Currently he is a Ph.D. candidate in School of Computer Science, Fudan University, Shanghai. His research interests include natural language processing and deep learning.



Xi-Peng Qiu received his B.S. degree in chemistry, and his Ph.D. degree in computer science from Fudan University, Shanghai, in 2001 and 2006 respectively. Currently he is a professor in School of Computer Science, Fudan University, Shanghai. His research interests include natural language processing and deep learning.

Xuan-Jing Huang received her B.S. and Ph.D. degrees in computer science from Fudan University, Shanghai, in 1993 and 1998 respectively. Currently she is a professor in School of Computer Science, Fudan University, Shanghai. Her research interests include natural language processing and deep learning.