Improving Clinical Named Entity Recognition with Global Neural Attention

Guohai Xu, Chengyu Wang, and Xiaofeng He(✉)

School of Computer Science and Software Engineering,
East China Normal University, Shanghai, China
guohai.explorer@gmail.com, chywang2013@gmail.com, xfhe@sei.ecnu.edu.cn

Abstract. Clinical named entity recognition (NER) is a foundational technology to acquire the knowledge within the electronic medical records. Conventional clinical NER methods suffer from heavily feature engineering. Besides, these methods treat NER as a sentence-level task and ignore the long-range contextual dependencies. In this paper, we propose an attention-based neural network architecture to leverage document-level global information to alleviate the problem. The global information is obtained from document represented by pre-trained bidirectional language model (Bi-LM) with neural attention. The parameters of pre-trained Bi-LM which makes use of unlabeled data can be transferred to NER model to further improve the performance. We evaluate our model on 2010 i2b2/VA datasets to verify the effectiveness of leveraging global information and transfer strategy. Our model outperforms previous state-of-the-art method with less labeled data and no feature engineering.

Keywords: Clinical named entity recognition · Neural attention
Language model

1 Introduction

The clinical text in electronic medical records has the potential to make a significant impact in many aspects of healthcare research such as drug analysis, disease inference, clinical decision support, and more. To analyze such clinical free text, one sequence labeling application namely NER plays a crucial role to identify medical entities at first step. Table 1 shows a clinical snippet containing such medical entities.

NER is still a challenging task in the clinical domain due to the distinctive characteristics of language. Dictionary-based methods fail to tag abbreviated phrases and acronyms which are common in clinical text. Rule-based systems are laborious to implement and trend to miss a number of misspellings that have their specific meaning. To overcome these limitations, various machine learning algorithms have been proposed to improve the performance. However, traditional machine learning approaches rely heavily on hand-crafted features,

Y. Cai et al. (Eds.): APWeb-WAIM 2018, LNCS 10988, pp. 264–279, 2018.
https://doi.org/10.1007/978-3-319-96893-3_20

it is especially tough to design features in the clinical-specific domain where specialized knowledge is needed. In the past few years, due to the simple but effective pre-trained word embedding [3,20,23], neural network models with as input distributed word representations achieve competitive performance against traditional models. Thus, current sequence labeling models typically include a RNN-based network that encodes each token into context vector and a CRF layer that decodes the representation to make predictions [10,15,21].

Table 1. A snippet of clinical text containing medical concepts, such as disease entities (in red), test entities (in blue) and treatment entities (in green).

The patient is a 58 year old right hand dominant white male with a long history of *hypertension* and *adult onset diabetes mellitus*. On physical examination, patient is in no acute distress, afebrile, *blood pressure* 134/80, *heart rate* 80 and regular, no bruits.
The patient was managed with *Vasotec*, *Nifedipine* and *Clonidine* with *blood pressure* under good control at the time of discharge, average 125 *systolic*, 70 *diastolic*, *heart rate* of 72. The patient was started on 2.5 of *Micronase* with *resulting sugars* as low as 63, decreased to 1.25 mg q.day.

Above-mentioned methods in practice treat NER as a sentence-level task where sentences in the same document are viewed as independent. However, clinical documents which are generated by physician to record the process of patients' treatments are centered on one or a few diseases. As shown in Table 1, the medical entities are topic-related to describe the condition of patients, for example, "*Vasotec*" (treatment entity) is used to control the "*blood pressure*" (test entity) due to his "*hypertension*" (disease entity). Thus, the long-range contextual dependencies are useful to improve the performance of sentence-level NER methods. Besides, ignoring the long-range contextual dependencies will lead to tagging non-consistency problem that the same mentions separated in different sentences from a document are tagged with different labels.

In this paper, we propose an attention-based stacked bidirectional long short-term memory with conditional random field (Att-BiLSTM-CRF) for clinical named entity recognition. Our model leverages global information within document and makes use of unlabeled data to achieve better performance. Inspired by the work of Peters et al. [24], we first pre-train a word embedding model and a bidirectional neural language model (Bi-LM) on unlabeled corpus in unsupervised learning (Sect. 3.2). Thus, the pre-trained Bi-LM can represent the sentences from document containing the global information. Then, we adopt stacked BiLSTM to encode the input sentence which consists of word embeddings, and incorporate all the representation of sentences within the document which the input sentence in with neural attention (Sect. 3.3). Finally, we use a CRF layer [14] to decode the representations to make sequence decision. The main contributions of this paper can be summarized as follows:

- We propose an attention-based neural network architecture namely Att-BiLSTM-CRF to incorporate global information to alleviate the problem of ignoring long-range contextual dependencies for clinical NER task.
- We transfer the parameters of pre-trained Bi-LM which makes use of unlabeled data to BiLSTM and show the advantages of transfer strategy than random initialization.
- Combining the global neural attention and pre-trained Bi-LM, our model outperforms previous state-of-the-art method on 2010 i2b2/VA datasets [25] with less labeled data and no feature engineering.

The rest of this paper is organized as follows: Sect. 2 discusses related research. Section 3 formulates the task and describes the architecture. Section 4 describes the datasets, training, experiments and results. Section 5 summarizes the paper.

2 Related Work

Our method is based on two lines of research which are sequence labeling and how to improve it with global information. Therefore, we mainly outline the recent work on NER and previous efforts in clinical domain. Then we will review the related work which aims to capture global information.

2.1 Named Entity Recognition

NER is a widely studied sequence labeling task, and many different approaches have been proposed. Among them, neural network models have been rapidly growing in popularity as they can be trained end-to-end with no feature engineering and task-specific resources. Taking inspiration from research of feed-forward network presented by Collobert et al. [3], Huang et al. [10] use a BiLSTM over a sequence of word embeddings and other hand-crafted spelling features with a CRF layer on top. Chiu and Nichols [4] also propose a similar model, but instead use CNN to learn character-level features. Lample et al. [15] also employ a similar architecture, but utilize LSTM to learn character-level features instead. Similar to Chiu and Nichols [4], Ma and Hovy [21] also use CNN to model character-level information, but without using any data preprocessing and achieving better NER performance. To relieve the limitation of relatively little labeled data, Peters et al. [24] explore a general semi-supervised approach which uses pre-trained neural bidirectional LM to augment context sensitive representation from large unlabeled corpus to improve previous methods.

Our architecture is based on the success of BiLSTM-CRF model [10,15,21], and is further modified to better incorporate global information with neural attention. Our model employs stacked BiLSTM to effectively model the context and excluding character-level information for simplicity. Furthermore, the Bi-LM can make use of unlabeled data and a simple transfer strategy can further improve the performance.

In clinical domain, there are a number of traditional machine learning algorithms based on hand-crafted features and domain-dependent knowledge or resources. Uzuner et al. [25] overview performance of systems on 2010 i2b2/VA challenge in detail. Among the all submitted systems in the evaluations, de Bruijn et al. [6] ranked first, and they trained a hidden semi-Markov model based on unsupervised feature representations obtained by Brown clustering and other text-oriented features. Subsequent work can be roughly divided into two directions. On the one hand, researchers focus attention on better feature representations. Jonnalagadda et al. [11] explore the use of distributed semantics derived empirically from unannotated text to improve the performance of clinical NER. Wu et al. [26] systematically compare two word neural embedding algorithms and show that low-cost distributed feature representations can be better than Brown clustering. On the other hand, researchers concentrate on appropriate data-preprocessing. Fu and Ananiadou [8] show that truecasing and annotation combination can best increase the NER system performance. Boag et al. [1] develop a lightweight tool by cascading CRF and SVM classifiers for clinical NER. Until recently, Chalapathy et al. [5] explore the effectiveness of BiLSTM-CRF based on off-the-shelf word embedding without any hand-crafted features. In contrast, the most advantage of our architecture is requiring no task-specific knowledge or feature engineering, and meanwhile achieving better performance with augmented global information.

2.2 Leveraging Global Information

Several studies have noticed the importance of global information to aid sentence-level NER. Finkel et al. [7] take non-local information into account while preserving tractable inference with Gibbs sampling. Krishnan and Manning [13] propose a two-stage model for exploiting non-local dependency. They use first CRF-based NER model using local features to make predictions and then train second CRF based on the output of the first CRF to maintain label consistency. Recently, Liu et al. [16] propose an extension to CRFs by integrating external memory to capture long-range contextual dependencies. Luo et al. [18] regard the whole document as input into BiLSTM-based NER model with self-attention mechanism. However, the method is only effectively applied to short text because RNN-based (including LSTM) models perform poorly as the length of input sentence increases [2,17].

Inspired by these earlier work, we also leverage global information to improve performance of clinical NER. In contrast, we propose a neural network architecture to combine the local and global information with neural attention. The stacked BiLSTM has its advantage over encoding sequential inputs than plain linear-chain CRF based on hand-crafted features. The performance of our method is not suffer from the variant length of document.

3 Neural Network Architecture

In this Section, we first provide the task definition and flow of our method for the problem of clinical NER. Then we illustrate the approach to pre-train the word embedding model and Bi-LM which is a key component in our architecture. Finally, we describe attention-based neural network architecture from bottom to up in detail.

3.1 Overview

Task Definition. We formally describe the Clinical NER task as follows: Given a sentence, $s = (w_1, w_2, ..., w_n)$ where n is the length of the sentence, find the medical entities $o = (y_1, y_2, ..., y_n)$ where y is the predefined label. The problem is a typical sequence labeling task. We use the BIO format to tag the entities. In particular, there are three medical entity categories: Disease (Dise for short), Test (Test for short), Treatment (Trea for short). If the word is the first word in medical entities, the word is labeled B-X (X is the entity category). The word is labeled I-X if the word is inside but not the first position of the medical entities. Otherwise, the word is labeled O.

For instance, which is shown in Fig. 1, the input sentence is (*a*, *long*, *history*, *of*, *hypertension*), then the model can output the sequence tag (O, O, O, O, B-Dise).

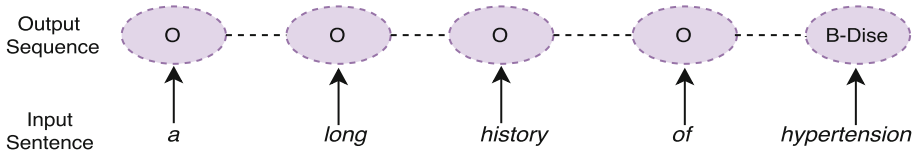


Fig. 1. An example for sequence labeling task.

In contrast to previous work, we additionally leverage the global information from the document $D = (s_1, s_2, ..., s_m)$ where the input sentence s is located to improve the performance. Thus, all of the representation of sentences in document will be utilized to complement the single input sentence. In a nutshell, the input to our model not only contains the single sentence, but also incorporates all of the sentences from the same document.

Flow of the Method. As illustrated in Fig. 2, the main components in our architecture are *Pre-Training*, *Encoder*, *Neural Attention*, *Decoder* respectively. First of all, we use unlabeled corpus to pre-train word embedding model and Bi-LM. Secondly, the first BiLSTM takes the word embeddings of single sentence as input, and then the pre-trained Bi-LM represents all the sentences within the same document which the input sentence in. Next, the second BiLSTM integrates

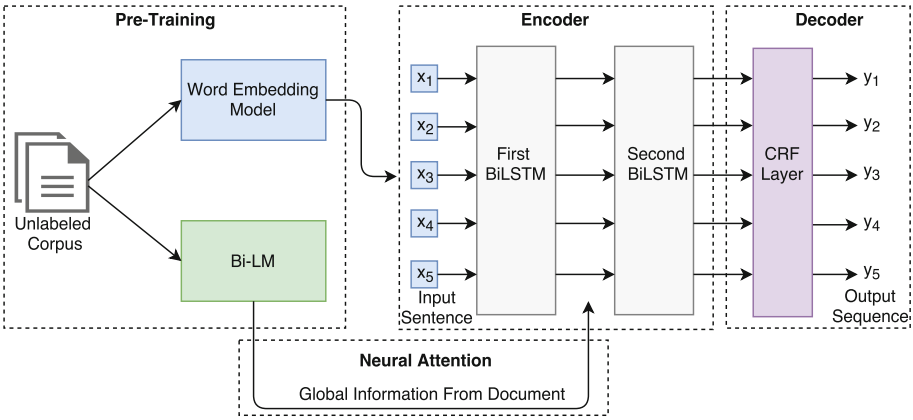


Fig. 2. The flow of our method for clinical NER.

the outputs of first BiLSTM and representations from Bi-LM that includes the global information from document with neural attention. Lastly, the CRF layer plays a decoding role to make sequence decision over the encoding of input.

3.2 Pre-training

Word Embedding Model. Word embedding is ubiquitous in NLP tasks since Mikolov et al. [20] propose an efficient method called Word2Vec for learning distributed representation of words. It is commonly believed that the word embedding captures useful semantic and syntactic information. Therefore, we use skip-gram algorithm [20] to train word embedding as input instead of heavily hand-crafted features.

Bi-LM. The Bi-LM is a vital component in our neural network architecture. On the one hand, pre-trained Bi-LM encodes the representation of sentences to enable the BiLSTM to look beyond the local context of sentence and extent to the global context of document. On the other hand, Bi-LM can make use of unlabeled data and its learned parameters can be transferred to first BiLSTM in NER model to improve performance. Now we describe the Bi-LM in detail.

Language model is proposed to learn a probability distribution over sequences of token pertaining to a language. Instead of count-based N-grams language model, we choose neural language model which has been shown to better retain long term dependencies. We use LSTM to model joint probabilities over word sequences which represented by word embeddings. Give a word sequence $(w_1, w_2, ..., w_n)$, LM computes the probability of the next word given all the previous words at each step. Here it can be called forward LM since we obtain the next word depending on the forward words, and LSTM is called forward LSTM as well. Thus, the overall probability can be written as:

$$p(w_1, w_2, ..., w_n) = \prod_{i=2}^{n} p(w_i | w_1, w_2, ..., w_{i-1}) \tag{1}$$

At each step, forward LSTM encode the history $(w_1, w_2, ..., w_{i-1})$ into a fixed dimensional vector $\overrightarrow{\mathbf{h}}_{i-1}^{LM}$ which is the hidden state of forward LSTM at position $i-1$ actually. Then, a softmax layer predicts the probability of next word w_i in the vocabulary. We train the forward LM model which maximizes the likelihood of given sentences in corpus.

A backward LM can be implemented in an analogous way if we reverse the word sequence. Thus, we obtain the similar overall probability:

$$p(w_n, w_{n-1}, ..., w_1) = \prod_{i=n-1}^{1} p(w_i | w_n, w_{n-1}, ..., w_{i+1}) \tag{2}$$

The backward LM predicts the previous word given the future sequence. Also, we utilize a backward LSTM to build the backward LM.

The forward and backward LSTM share the same input layer (word embedding layer) and output layer (softmax layer). After pre-training, the pre-trained Bi-LM can be used to represent sentences of document in training corpus. We concatenate the last cell state of forward and backward LSTM to represent the input sentence, i.e., $\mathbf{s} = [\overrightarrow{\mathbf{c}}_n^{LM}; \overleftarrow{\mathbf{c}}_1^{LM}]$.

Transfer Strategy. In NLP, pre-trained word embedding like Word2Vec [20] and GloVe [23] has been common initialization for the input layer of neural network models. The word vectors obtained from training on large amounts of unlabeled corpus achieve better performance than random initialization on a variety of NLP tasks. However, the form of transfer learning is not limited to word vectors, but also includes weights from pre-trained recurrent neural networks [22,27].

Inspired by above ideas, we propose a transfer strategy to further improve the performance of NER model. We let Bi-LM and first BiLSTM in Encoder component of NER model have the same architecture. Therefore, the parameters of pre-trained Bi-LM can be shared to the first BiLSTM. The well-trained Bi-LM from large, unlabeled corpus can help the NER model have a better initialization, thus leads to better performance.

3.3 Att-BiLSTM-CRF Model

Encoder. As depicted in Fig. 3, this architecture is similar to the ones presented by Huang et al. [10], Lample et al. [15] and Ma et al [21]. In contrast, we use stacked BiLSTM to encode sequential input for incorporating global information with neural attention.

For a given sentence $s = (w_1, w_2, ..., w_n)$ containing n words in a document $D = (s_1, s_2, ..., s_m)$ including m sentences. At first, the sentence is represented as a sequence of vectors $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_t, ..., \mathbf{x}_n)$ through the embedding layer.

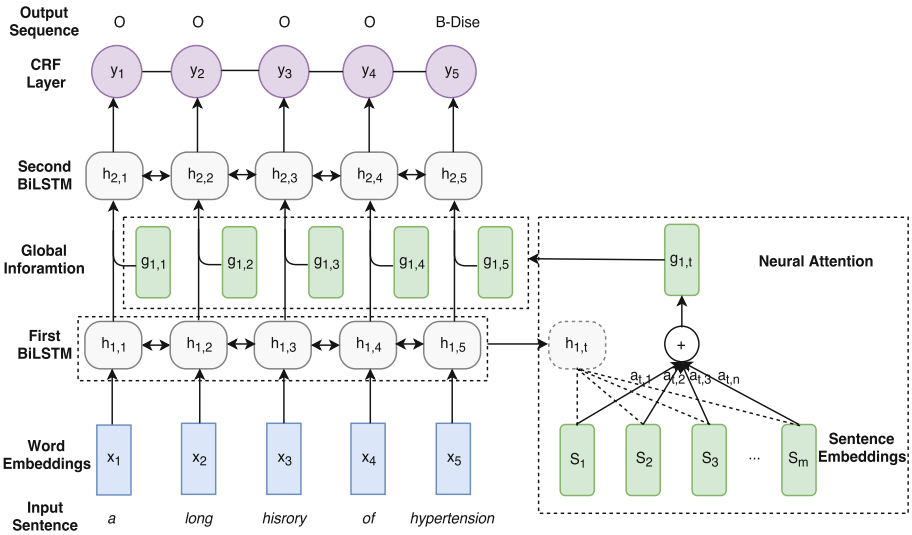


Fig. 3. The architecture of our Att-BiLSTM-CRF model.

Next, a forward LSTM in first BiLSTM computes a representation $\overrightarrow{\mathbf{h}}_{1,t}$ of the left context of the sentence at each word t, and a backward LSTM computes a representation $\overleftarrow{\mathbf{h}}_{1,t}$ of the same sequence in reverse. Then, the representation of each word t is obtained by concatenating its left and right context representations, $\mathbf{h}_{1,t} = [\overrightarrow{\mathbf{h}}_{1,t}; \overleftarrow{\mathbf{h}}_{1,t}]$.

In most previous NER methods, the representation of each word will be followed by a transformation layer and CRF layer to make prediction without considering the long-range contextual dependencies. While we introduce the Neural Attention component to leverage all the sentences in the document D. We use pre-trained Bi-LM to represent all the sentences which can be regard as global context. Then we apply the neural attention to seek the related global context based on the representation of each word which can be regard as local context. The global context in the document can supply extra useful information to each word. As a result, the extended representation of each word consists of the local context in sentence and the global context in document.

Every sentence in document D can be represented by pre-trained Bi-LM, thus we get a another sequence of vectors $\mathbf{D} = (\mathbf{s_1}, ..., \mathbf{s_j}, ..., \mathbf{s_m})$ for sentences. Firstly, we use an attention matrix A to calculate the similarity between the local context in sentence and global context in the document. The attention weight value $a_{t,j}$ in attention matrix A is computed by comparing the local context $\mathbf{h}_{1,t}$ with each sentence embedding $\mathbf{s}_j$:

$$a_{t,j} = \frac{exp(score(\mathbf{h}_{1,t}, \mathbf{s}_j))}{\sum_k exp(score(\mathbf{h}_{1,t}, \mathbf{s}_k))} \tag{3}$$

Above *score* is referred as a bilinear function which is borrowed from Bahdanau et al. [2] and Luong et al. [19]:

$$score(\mathbf{h}_{1,t}, \mathbf{s}_j) = \mathbf{h}_{1,t}^T \mathbf{W_a} \mathbf{s}_j \tag{4}$$

here the weight matrix $\mathbf{W_a}$ is a parameter of the model. Secondly, the global context $\mathbf{g}_{1,t}$ is computed as a weighted sum of each sentence embedding $\mathbf{s}_j$:

$$\mathbf{g}_{1,t} = \sum_{j=1}^{m} a_{t,j} \mathbf{s}_j \tag{5}$$

Thirdly, we concatenate the global context and local context into a vector $[\mathbf{h}_{1,t}; \mathbf{g}_{1,t}]$ to represent each word. Next, the extended representation of each word become a sequential of intermediate representation, which can be sent into second BiLSTM.

Decoder. After process of encoding, it is simple to use a linear layer to predict a score for each possible label independently based on the output of the second BiLSTM. But there are strong dependencies across output labels, for example, I-Dise cannot follow B-Test. Therefore, instead of modeling tagging decisions independently, we add another CRF layer to decode the best label path in all possible label paths. Followed by Lample et al. [15], we only consider the relations between labels in neighborhoods and jointly decode the best chain of labels.

We consider $\mathbf{P} \in \mathbb{R}^{n \times k}$ to be the matrix scores output by the second BiLSTM, where the n is length of input sentence and the k is the number of distinct labels. The element $P_{i,j}$ in the matrix is the score of j^{th} label of the i^{th} word in the sentence. We introduce a label transition matrix $\mathbf{T}$, where element $T_{i,j}$ represents a score of a transition from the label i to label j. After that, the whole input sentence $\mathbf{X}$ gets a sequence of predictions $\mathbf{y} = (y_1, y_2, ..., y_n)$ from model, we can define its score to be

$$s(\mathbf{X}, \mathbf{y}) = \sum_{i=1}^{n} (T_{y_{i-1}, y_i} + P_{i, y_i}) \tag{6}$$

where the transition matrix $\mathbf{T} \in \mathbb{R}^{(k+2) \times (k+2)}$ is the parameter of our model. In above equation, y_0 and y_n are the start and end labels of a given sentence. Therefore, the transition matrix $\mathbf{T}$ is a square matrix of size $k + 2$.

During training, we use the maximum conditional likelihood estimation. First, as shown in Eq. (7), a softmax function is used to normalize the above score over all possible label paths $\tilde{\mathbf{y}}$ to form the conditional probability of the path $\mathbf{y}$. Then, the log-likelihood of the conditional probability of the correct tag sequence is given in Eq. (8). We train the model to maximize the log-likelihood of the probability of all the correct tag sequences in labeled data to obtain the final parameters.

$$p(\mathbf{y}|\mathbf{X}) = \frac{exp(s(\mathbf{X}, \mathbf{y}))}{\sum_{\tilde{\mathbf{y}}} exp(s(\mathbf{X}, \tilde{\mathbf{y}}))} \tag{7}$$

$$\mathcal{L} = log(p(\mathbf{y}|\mathbf{X})) \tag{8}$$

During inference, as given in Eq. (9), the best label path $\mathbf{y}^*$ is predicted through computing the maximum score among all the possible label paths. Because we only consider the interactions between two successive labels, dynamic programming such as Viterbi algorithm can be applied to effectively computes the scores.

$$\mathbf{y}^* = argmax_{\tilde{\mathbf{y}}}\ s(\mathbf{X}, \tilde{\mathbf{y}}) \tag{9}$$

4 Experiments

4.1 Datasets

In this paper, we use datasets from 2010 i2b2/VA Natural Language Processing Challenges for Clinical Records[1] containing a concept extraction task focused on identifying medical concepts from realistic clinical narratives. Because of the restrictions introduced by Institutional Review Board (IRB), only part of original datasets is available. The challenge requires the systems to predict the exact boundary of medical concepts and classify them into specified category including problem, test, treatment and other. Table 2 summarizes the statistics of labeled datasets which we have used in our experiments. In addition, we get a number of unlabeled clinical notes from MIMIC-III corpus[2] [12] for pre-training word embedding model and Bi-LM.

Table 2. A basic statistics of datasets.

	Training data	Test data	Unlabeled data
# Documents	170	256	5000
# Sentences	16315	27626	1042534
# Mentions	16525	31161	-

4.2 Model Training

Preparation and Evaluation. We split the training data into two parts, 130 documents (about 80%) for training set and 40 documents (about 20%) for development set. We tune the hyperparameters of our model on development set and report the results on the test set. Note that, to compare to other existing methods (Sect. 4.5), the final training is done on both the training and development sets. We don't do any feature engineering except using a special token for numbers. For evaluation, we do exact matching of entity mentions to compute micro-precision, micro-recall and micro-F_1.

[1] https://i2b2.org/NLP/DataSets/Main.php.
[2] https://mimic.physionet.org.

Model Architecture Details. Dimensions of word embedding are set 300. For language model, the hidden state of LSTM has 300 dimensions. For first BiLSTM in NER model, the hidden state of LSTM also has 300 dimensions. In consideration of the transfer strategy, the first BiLSTM and Bi-LM have identical parameter setting. For second BiLSTM in NER model, as it concatenates the output of first BiLSTM and the representations of global information, the dimensions of the hidden state of LSTM are 600.

Training Details. For word embedding model, we use skip-gram algorithm [20] to obtain word vectors on unlabeled data. For Bi-LM, the input embedding layer is initialized with the weights from word embedding model and other parameters are initialized with Xavier initialization [9]. Once the pre-training is done, we use pre-trained Bi-LM to represent the sentences in document and the parameters of Bi-LM also can be transferred to first BiLSTM in NER model. For NER model, the input embedding layer is also initialized with the weights from word embedding model and other parameter are initialized with Xavier initialization as well. We use SGD with momentum of 0.9 to train the NER model. We train our networks using back-propagation algorithm updating parameter on a batch size of 10. The initial learning rate is 0.01 and decay the learning rate by multiplying it by 0.9 if the F_1 score does not improve on development set for one epoch. We use a gradient clipping of 5.0 to avoid gradient exploding problem. We train the model for 30 epochs and use early stopping to avoid over-fitting.

4.3 Effectiveness of Leveraging Global Information

In this part, we verify the effectiveness of global neural attention augmented BiLSTM (Att-BiLSTM) compared with plain BiLSTM. In previous work, most methods treat NER as a sentence-level task. In Contrast, we incorporate the global information in document to capture the long-range contextual dependencies. As shown in Table 3, in irrespective of the impact of CRF, we perform the contrast experiments based on only BiLSTM to evaluate the ability of presentations for each input word. From the results, we see that the number of layers affects the performance. In both of Att-BiLSTM and BiLSTM, the stacked BiLSTM outperforms the BiLSTM with single layer. Also, the global neural attention gives an improvement over the plain BiLSTM due to the leveraging of global information. We observe that the F_1 of stacked Att-BiLSTM is 81.62%, which is an absolute improvement of 1.01% over the plain stacked BiLSTM with no global neural attention.

To be honest, global neural attention don't show obvious effects when the Att-BiLSTM only has one layer. It is because the final tagging predictions mainly depend on local context for each word, while global context only supplements extra information. Therefore, our model need another layer to encode the sequential intermediate vectors containing global context and local context. In other words, the architecture needs second BiLSTM to learn the differences between the two contexts.

Table 3. Performance of leveraging global information.

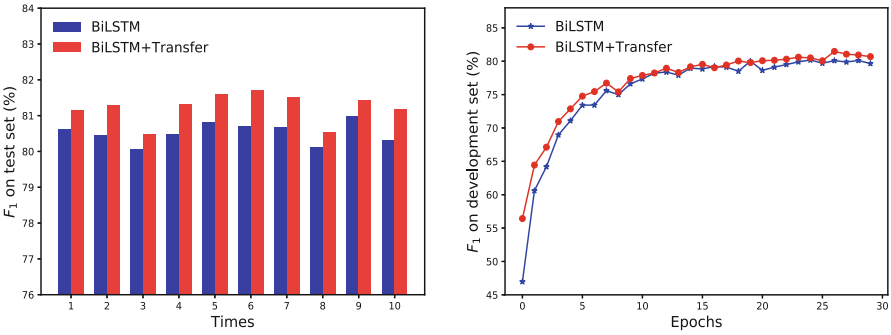
Model	Layers	Precision (%)	Recall (%)	F_1(%)
BiLSTM	1	77.53	81.63	79.53
	2	80.50	80.71	80.61
Att-BiLSTM	1	78.25	81.17	79.68
	2	80.62	82.65	**81.62**

4.4 Effectiveness of Transfer Strategy

In this part, we verify the effectiveness of transfer strategy. The baselines are stacked BiLSTM and stacked Att-BiLSTM obtained from above experiments. In baseline methods, we initialize the parameters of their stacked BiLSTM with Xavier initialization [9] which has been regarded as an effective initialization strategy. In comparison to Xavier initialization, we initialize the parameters of first BiLSTM from the parameters of pre-trained Bi-LM. The results is showed in Table 4, the simple transfer strategy gives an additional improvement over baselines. For stacked BiLSTM, the F_1 gets an absolute improvement of 0.53%. Also for stacked Att-BiLSTM, the absolute improvement is 0.71% in F_1 score.

Table 4. Performance of transfer strategy.

Model	Transfer	Precision (%)	Recall (%)	F_1(%)
BiLSTM	No	80.50	80.71	80.61
	Yes	80.16	82.15	81.14
Att-BiLSTM	No	80.62	82.65	81.62
	Yes	81.58	83.08	**82.33**



(a) Variance. (b) Convergence.

Fig. 4. Comparison between plain stacked BiLSTM and stacked BiLSTM with transfer strategy.

To further verify the effectiveness of transfer strategy, we train the model with different random seeds. At first, we respectively train the stacked BiLSTM and stacked BiLSTM with transfer strategy for ten times in different random seeds. From the results in Fig. 4(a), it shows that the transfer strategy always increases the performance of plain stacked BiLSTM more or less. We compute the mean F_1 score of stacked BiLSTM with transfer strategy is 81.22%, and its variance is 0.15%. In contrast, the mean F_1 score of plain BiLSTM is only 80.52%, and its variance is 0.08%. Then we randomly select one example to draw the convergence of the two models. As depicted in Fig. 4(b), transfer strategy accelerates the model training especially at the first several epochs. Also the transfer strategy helps the model achieve the better performance at last. Above comparisons prove the effectiveness of transfer strategy, we believe that it can promote other similar models which contain LSTM.

4.5 Comparison to Other Methods

In this part, we compare the performance of our model with other existing methods on the 2010 i2b2/VA datasets. The results are shown in Table 5, the name of other methods followed by Chalapathy et al. [5]. We have implied the main ideas of other methods in related work (Sect. 2.1). Form the results, our model obtains the state-of-the-art performance than others. Although we only get nearly 0.5% F_1 score higher than the previous state-of-the-art method which is the best submission from the 2010 i2b2/VA challenge, their model is based on original dataset which has more than twice labeled data than ours.

To understand the importance of leveraging global information and transfer strategy, we implement the common BiLSTM-CRF model as baseline. The results confirm that leveraging global information increases F_1 score by 0.53% (from 84.66% to 85.19%) and increases F_1 score by 1.05% (from 84.66% to 85.71%) with additional transfer strategy. We conclude that our model relieves

Table 5. Performance comparison with other existing methods on the 2010 i2b2/VA datasets. * indicates models trained with the use of original larger labeled data.

Model	Precision (%)	Recall (%)	F_1(%)
Distributional semantics CRF * [11]	85.60	82.00	83.70
Hidden semi-markov model * [6]	86.88	83.64	**85.23**
Truecasing CRFSuite [8]	80.83	71.47	75.86
CliNER [1]	79.50	81.20	80.00
Binarized neural embeding CRF [26]	85.10	80.60	82.80
Glove-BiLSTM-CRF [5]	84.36	83.41	**83.88**
BiLSTM-CRF	86.21	83.17	84.66
Att-BiLSTM-CRF	85.51	84.87	85.19
Att-BiLSTM-CRF + Transfer	86.27	85.15	**85.71**

the problem of ignoring the long-range contextual dependencies and the pre-trained Bi-LM makes use of unlabeled data to further improve the performance.

5 Conclusion

In this paper, we propose an attention-based neural network architecture to leverage document-level global information to alleviate the problem of ignoring long-range contextual dependencies for clinical NER task. In addition, we explore a transfer strategy to further make use of unlabeled data using pre-trained Bi-LM. Our results of experiments show that the transfer strategy consistently improve the performance. Owing to the above two advantages, our model achieves the state-of-the-art performance on public 2010 i2b2/VA datasets.

Although we use clinical data to verify the effectiveness of our method, the Att-BiLSTM-CRF model can be adapted to other domain where global context is useful. Moreover, the transfer strategy using Bi-LM has generalization performance.

Acknowledgments. This work was supported by the National Key Research and Development Program of China under Grant No. 2016YFB1000904.

References

1. Boag, W., Wacome, K., Naumann, T., Rumshisky, A.: CliNER: a lightweight tool for clinical named entity recognition. AMIA Joint Summits on Clinical Research Informatics (poster) (2015)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. In: Proceedings of the 3rd International Conference on Learning Representations (2015)
3. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. J. Mach. Learn. Res. **12**, 2493–2537 (2011)
4. Chiu, J.P.C., Nichols, E.: Named entity recognition with bidirectional LSTM-CNNs. In: Proceedings of TACL, pp. 357–370 (2016)
5. Chalapathy, R., Borzeshi, E.Z., Piccardi, M.: Bidirectional LSTM-CRF for clinical concept extraction. In: Proceedings of the Clinical Natural Language Processing Workshop ClinicalNLP, pp. 7–12 (2016)
6. de Bruijn, B., Kiritchenko, C.C., Martin, J.D., Zhu, X.D.: Machine-learned solutions for three stages of clinical information extraction: the state of the art at i2b2 2010. J. Am. Med. Inf. Assoc. **18**(5), 557–562 (2011)
7. Finkel, J.R., Grenager, T., Mannning, C.D.: Incorporating non-local information into information extraction systems by Gibbs sampling. In: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics, pp. 363–370 (2005)
8. Fu, X., Ananiadou, S.: Improving the extraction of clinical concepts from clinical records. In: Proceedings of the 4th Workshop on Building and Evaluating Resources for Health and Biomedical Text Processing (2014)

9. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, pp. 249–256 (2010)
10. Huang, Z.H., Xu, W., Yu, K.: Bidirectional LSTM-CRF models for sequence tagging. CoRR, abs/1508.01991 (2015)
11. Jonnalagadda, S., Cohen, T., Wu, S.T., Gonzalez, G.: Enhancing clinical concept extraction with distributional semantics. J. Biomed. Inf. **45**(1), 129–140 (2012)
12. Johnson, A.E., Pollard, T.J., Shen, L., Li-wei, H.L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L.A., Mark, R.G.: MIMIC-III, a freely accessible critical care database. Sci. Data **3**, 160035 (2016)
13. Krishnan, V., Manning, C.D.: An effective two-stage model for exploiting non-local dependencies in named entity recognition. In: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, pp. 1121–1128 (2006)
14. Lafferty, J.D., McCallum, A., Pereira, P.: Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proceedings of the 18th International Conference on Machine Learning, pp. 282–289 (2001)
15. Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural architectures for named entity recognition. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 260–270 (2016)
16. Liu, F., Baldwin, T., Cohn, T.: Capturing long-range contextual dependencies with memory-enhanced conditional random fields. In: Proceedings of the Eighth International Joint Conference on Natural Language Processing, pp. 555–565 (2017)
17. Lai, S.W., Xu, L.H., Liu, K., Zhao, J.: Recurrent convolutional neural networks for text classification. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, pp. 2267–2273 (2015)
18. Luo, L., Yang, Z.H., Yang, P., Zhang, Y., Wang, L., Lin, H.F., Wang, J.: An attention-based BiLSTM-CRF approach to document-level chemical named entity recognition. Bioinformatics **1**, 8 (2017)
19. Luong, T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1412–1421 (2015)
20. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26th Annual Conference on Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
21. Ma, X.Z., Hovy, E.H.: End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pp. 1064–1074 (2016)
22. McCann, B., Bradbury, J., Xiong, C.M., Socher, R.: Learned in translation: contextualized word vectors. In: Proceedings of the 30th Annual Conference on Advances in Neural Information Processing Systems, pp. 6297–6308 (2017)
23. Penningto, J., Socher, R., Manning, C.D.: GloVe: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pp. 1532–1543 (2014)
24. Peters, M.E., Ammar, W., Bhagavatula, C., Power, R.: Semi-supervised sequence tagging with bidirectional language models. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, pp. 1756–1765 (2017)

25. Uzuner, O., South, B.R., Shen, S.Y., DuVall, S.L.: 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. J. Am. Med. Inf. Assoc. **18**(5), 552–556 (2011)
26. Wu, Y.H., Xu, J., Jiang., M., Zhang., Y.Y., Xu, H.: A study of neural word embeddings for named entity recognition in clinical text. In: Proceedings of the 2015 American Medical Informatics Association Annual Symposium, pp. 1326–1333 (2015)
27. Yang, Z.L., Salakhutdinov, R., Cohen, W.W.: Transfer learning for sequence tagging with hierarchical recurrent networks. In: Proceedings of the 5th International Conference on Learning Representations (2017)