Clinical Named Entity Recognition via Bi-directional LSTM-CRF Model

Jinhang Wu, Xiao Hu, Rongsheng Zhao, Feiliang Ren*, Minghan Hu

School of Computer Science and Engineering, Northeastern University, Shenyang, 110819, China
*Corresponding Author: renfeiliang@mail.neu.edu.cn

Abstract. EMR (Electronic Medical Record) refers to the systematized collections of patients' electronically-stored health information in a digital format. Clinical Named Entity Recognition aims to recognize and extract entity mentions from EMR. In this paper, we introduce a novel neural network architecture based on bidirectional LSTMs and conditional random fields, requiring no massive hand-crafted features or data pre-processing, compared to traditional statistical methods such as HMM and CRF.

Keywords: named entity recognition; LSTM; conditional random field; medical text processing

1 Introduction

Named entity recognition (NER) is one of the most important tasks for development of more sophisticated NLP systems. For news text, NER task has achieved relatively good performance. However, in other domains such as medical domain, there is still large gap. The main reasons for this gap are as follows: complicate and inconsistent terminologies, and ambiguities caused by abbreviations and acronyms.

CCKS2017 task2 is named entity recognition for electronic medical records, referred to as CNER. For a given group of electronic medical records (text file), the task is to identify and extract the related medical clinical entity names (entity mentions), and they are classified into pre-defined categories, including disease, symptom, examination, treatment and body part.

In numerous means of NER, CRF and LSTM are widely used. CRF (conditional random field) is a conditional probability model for marking ordered data, which combines the characteristics of the maximum entropy model and the HMM model. However, it can't take the long term contextual information into account. LSTM model, which is powerful in sequence modeling, can capture long term context information.

According to this, in this paper, we present a novel model based on bidirectional LSTMs and conditional random fields, which provides the best NER results ever reported in standard evaluation settings, even compared with models that use external resources, such as gazetteers.

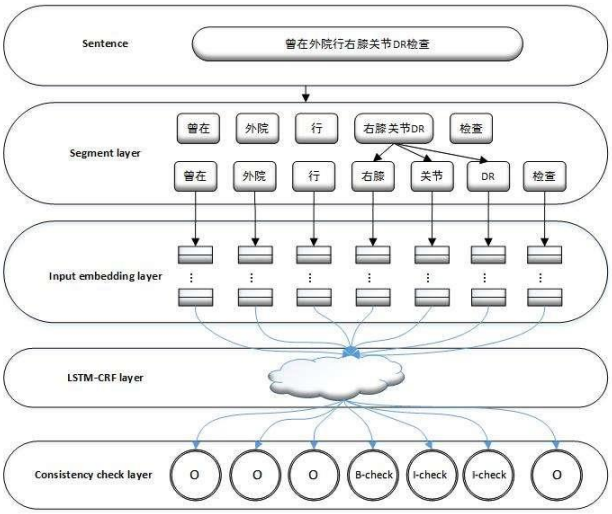

Figure 1:Architecture of the model

2 Model

We use a LSTM-CRF model (Guillaume Lample,Miguel Ballesteros,Sandeep Subramanian,Kazuya Kawakami,Chris Dye.2001) to implement NER for Electronic Medical Records. As shown in Figure 1, the model proposed in this paper contains four components:

(1) Segment layer: segment sentence and use BIO to mark the format required for the model;

(2) Input Embeddings layer: map each word into a low dimension vector;

(3) LSTM-CRF layer: utilize BLSTM-CRF to mark each word. The architecture is shown in Figure 2;

(4) Consistency check layer: check the consistency of the result of LSTM-CRF layer.

These components will be presented in detail in this section.

2.1 Segment

In Chinese, as is known to all, word boundaries are not readily identified in texts. Word segmentation is a key first step to generate features for an NER system. Unfortunately, the result of word segmentation always has various mistakes. For instance, in the following example, the most frequent mistake is that the context and part of mention combine together. For example, in the labeled data set, word '上腹部' is a mention, but character '上' connects with its left context.

"以上/腹部/疼痛/为/主要/症状/。"

To solve this problem, we investigate a new segmentation method named "ReSeg". Firstly, we find out all of the clinical named entities appearing in the training dataset, and keep these mentions in sentence not being segmented. Then on this result, we

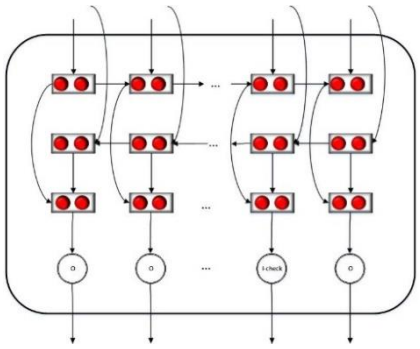

Figure 2:LSTM-CRF

segment these mentions again and require the final word segmentation result. And we'll analyze the effect of "ReSeg" in the experiments section.

we use the BIO format (Beginning, Inside, Outside) to represent sentences where every token is labeled as B-label if the token is the beginning of a mention, I-label if it is inside a mention but not the first token within the mention, or O which means otherwise. Furthermore, we follow the same format with the CoNLL2003 sharing task as the final input format for the training: each word has to be on a separate line, and there must be an empty line after each sentence. A line must contain at least two columns, the first one being the word itself, the last one being the named entity.

2.2 Input Embeddings

Our input embedding contains two parts: character embeddings and word embeddings. We use pre-trained word embeddings to initialize our lookup table. Embeddings are pre-trained using word2vec and Glove respectively. Word embeddings are trained based on the training dataset and the unlabeled dataset in CCKS-2017 shared task-2. In detail, the embedding dimension is 100, window size is 8 and iteration times is 100.

2.3 LSTM-CRF

Recurrent neural networks (RNNs) are a family of neural networks that operate on sequential data. Though, in theory, RNNs are capable of capturing long-distance dependencies, in practice, they fail due to the gradient vanishing or exploding problems.

Long short-term memory networks (LSTMs) are variants of RNNs designed to solve these gradient vanishing or exploding problems. Basically, a LSTM unit contains three multiplicative gates which control the proportions of information to forget and to pass on in the next time step. Formally, the formulas to update an LSTM unit at time *t* are:

$$\mathbf{i}_t = \boldsymbol{\sigma}(\mathbf{W}_{xi}\mathbf{x}_t + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{W}_{ci}\mathbf{c}_{t-1} + \mathbf{b}_i)$$

$$\mathbf{c}_t = (\mathbf{1} - \mathbf{i}_t) \odot \boldsymbol{c}_{t-1} + \mathbf{i}_t \odot \mathbf{tanh}(\mathbf{W}_{xc}\mathbf{x}_t + \mathbf{W}_{hc}\mathbf{h}_{t-1} + \mathbf{b}_c)$$

$$\mathbf{o_t} = \boldsymbol{\sigma}(\mathbf{W}_{xo}\mathbf{x}_t + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{W}_{co}\mathbf{c}_t + \mathbf{b}_o)$$

$$\mathbf{h_t} = \mathbf{o}_t \odot \mathbf{tanh}(\mathbf{c}_t)$$

Where σ is the element-wise sigmoid function, and $\odot$ is the element-wise product. $\mathbf{x}_t$ represents current input; $\mathbf{i}_t$ represents a input gate with corresponding weight matrix $\mathbf{W}_{xi}$, $\mathbf{W}_{hi}$, $\mathbf{W}_{ci}$, $\mathbf{b}_i$; $\mathbf{o_t}$ represents a output gate with corresponding weight matrix $\mathbf{W}_{xo}$, $\mathbf{W}_{ho}$, $\mathbf{W}_{co}$, $\mathbf{b}_o$; $\mathbf{h}_{t-1}$ represents the state generated in previous step; $\mathbf{c}_t$ represents current state.

The representation of a word using this model is obtained by concatenating its left and right context representations. These representations effectively include a representation of a word in context, which is useful for numerous tagging applications.

Instead of modeling tagging decisions independently, we model them jointly using a conditional random field (Lafferty et al., 2001).

2.4 Consistency check

Consistency check is a method of checking whether the predicted label class is consistent with the label class in training dataset. We have statistically evaluated 2399 different mention-label pairs in the training data. If the predicted label class is not the same as the original one in training dataset, we will manually modify the label to original label. For instance, the label of word "腔隙性脑梗死" in predicted sentence is "症状和体征"; but in training dataset this word is labeled as "疾病和诊断". In this case, we change the word's label to original label.

More formal, we denote the mention and label in predicted sentence by $pair_p$ = $(mention, label_p)$, and $pair_t$ = $(mention, label_t)$ in training dataset respectively.

$$\forall\, (pair_p.mention = pair_t.mention \wedge pair_p.label_p \neq pair_t.label_t)$$
$$\gg pair_p.label_p = pair_t.label_t$$

3 Experiments

3.1 Dataset

We test our model on the dataset provided by CCKS 2017 CNER task. These data are electronic medical records. There are four different file folders and each one contains 300 labeled files and 2605 unlabeled files.

We use these data to randomly get train dataset (80%) and dev dataset (10%). And we can also get test dataset (10%).

3.2 Baseline

In this section, we use the same dataset (CCKS-2017 shared task-2) and 4 different word segmentation tools to compare the performance of three common models without any pre-processing or feature engineering. One of models is traditional statistics

methods based CRF (conditional random field). One of them is the single neural network architecture –LSTM (Long Short Term Memory Unit). And the last one is the combination of two previous models of LSTM and CRF. In addition, we randomly initialize the word embeddings of LSTM.

Models	jieba	nlpir	Stanford	hanlp
CRF	79.95	80.17	80.95	82.01
LSTM	82.05	83.14	83.2	84.98
LSTM-CRF	83.18	84.36	84.93	86.28

Table 1: CNER F1 score with different models and word segmentation tools

In Table 1, LSTM-CRF model with *hanlp* tool obtains better performance than others. Therefore, we choose the LSTM-CRF with *hanlp* as the baseline model.

3.3 Results and Discussion

Error propagation is a difficult challege in NLP tasks, particularly in Chinese text processing. Word segmentation is a key first step to generate features for Chinese NER system. To reduce the mistakes created by word segmentation, we propose a method named "ReSeg".

Models	F1
ReSeg + CRF	83.89
ReSeg + LSTM	87.05
ReSeg + LSTM-CRF	88.19

Table 2: Impact of "ReSeg" for each model

Models	Variant	F1
ReSeg + LSTM-CRF	pretrain	91.47
ReSeg + LSTM-CRF	pretrain+dropout	92.9
ReSeg + LSTM-CRF	pretrain+dropout+char	91.13
ReSeg + LSTM-CRF + CC	pretrain+dropout+char	93.41

Table 3: NER results with our models, using different configurations. "pretrain" refers to models that include pretrained word embeddings, "char" refers to models that include character-based modeling of words, "dropout" refers to models that include dropout rate. "CC" refers to models that include Consistency Check.

In order to analyze the influence of "ReSeg", we use the "ReSeg" method for each model. Parameter settings are consistent with 3.2.

In Table 2, we observe that the performence of each model is improved after using "ReSeg". Especially on LSTM-CRF model it yields nearly 2% absolute improvement.

The result shows that “ReSeg” is indeed useful and has no model restrictions, in other words, every model could use the “ReSeg” method.

In Table 3, We observed that pretraining our word embeddings gave us the biggest improvement in overall performance of +3.28 in F1. The dropout gave us an increase of +1.43 and finally learning character-level word embeddings resulted in an increase of about +0.51. The Consistency Check layer gave us an increase of +2.28.

4 Conclusion

In this paper, we propose a novel LSTM-CRF method to solve named entity recognition for electronic medical records. A key aspect of our model is that we combine traditional statistical model with neural network. The proposed model is very robust and it achieves better performance without using any external resources.

5 Acknowledgements

This work is supported by the National Natural Science Foundation of China (NSFC No. 61572120, 61300097and 61432013). We thank all anonymous reviewers for their constructive comments.

6 References

1. Ronan Collobert, Jason Weston, Léon Bottou. 2011. Natural language processing (almost) from scratch.
2. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space.
3. Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality.
4. Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever. 2012. Improving neural networks by preventing co-adaptation of feature detectors.
5. John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.
6. Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, Chris Dyer. 2016. Natural Architectures for Named Entity Recognition.
7. Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging.
8. Yoon Kim, Yacine Jernite, David Sontag, and Alexander M. Rush. 2015. Character-aware neural language models.
9. Silviu Cucerzan and David Yarowsky. 1999. Language independent named entity recognition combining morphological and contextual evidence.