Clinical Named Entity Recognition Method Based on CRF

Yanxu Chen[1], Gang Zhang[1], Haizhou Fang[1], Bin He, and Yi Guan[⋆]

Research Center of Language Technology
Harbin Institute of Technology, Harbin, China

Abstract. Clinical name entity recognition is a task of CCKS2017. The purpose of this task is to recognize symptom, disease, exam, treatment and body words from medical records. In this paper, we propose two methods based on conditional random fields (CRFs) and LSTM-CRF. The experiment shows that our system is effective in the clinical name entity recognition of medical records, achieving a F_1 measure of 0.8974 at the strict entity evaluation level which ranked sixth.

Keywords: Name entity recognition,conditional random fields,Long Short-Term Memory

1 Task analysis

1.1 Task definition

For a given set of electronic medical records, the goal of the task is to identify and extract the entity mention related to medical clinics and classify them into pre-defined categories, such as symptom, disease, exam, treatment and body.

2 LSTM-CRF method

The neural networks is widely used to bulid the state-of-the-art sequence labeling systems. We chose the network architecture combining of bidirectional LSTM and CRF.

2.1 Basic model

The evaluation task gives the gold standard annotation data and the unlabeled data, given the entity location and category in the text. We use the char as a unit for sequence to modeling the text, deal with the entity recognition as a sequence labeling problem. Make this problem be a seq2seq model.

[⋆] Corresponding author at: Mailbox 321, West Da-zhi Street 92, Harbin, Heilongjiang, China. Tel.: +86 18686748550. E-mail address: guanyi@hit.edu.cn (Y. Guan)

[1] These three authors contribute equally to this study.

2.2 Neural Network Architecture

In this section, we describe the components (layers) of our neural network architecture. We introduce the neural layers in our neural network one by-one from bottom to top.

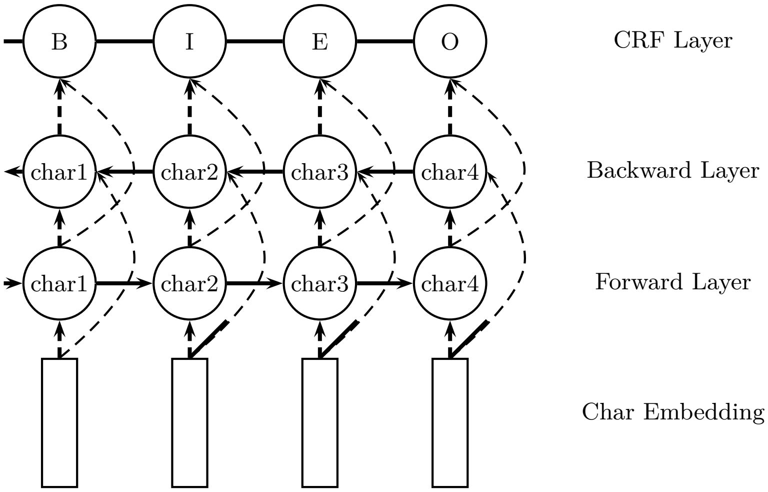

Fig. 1. neural network architecture

Char Embedding We use the char level vector to represent the Chinese words,by using the word2vec in the unlabeled dataset, the embedding dimension is 100.

Bi-directional-LSTM The basic LSTM is designed to cope with the gradient vanishing problems of RNN.The formulas to update LSTM unit at time t are:

$$i_t = \sigma(W_i h_{t-1} + U_i x_t + b_i) \tag{1}$$

$$f_t = \sigma(W_f h_{h-1} + U_f x_t + b_f) \tag{2}$$

$$\overline{c}_t = \tanh(W_c h_{t-1} + U_c x_t + b_c) \tag{3}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \overline{c}_t \tag{4}$$

$$o^t = \sigma(W_o h_{h-1} + U_o x_t + b_o) \tag{5}$$

$$h_t = o_t \odot \tanh(c_t) \tag{6}$$

The σ is the sigmoid function and $\odot$ is the convolution product. x_t is the input vector (e.g. embedding layer) at the time t, and the h_t is the hidden state vector storing the past information at(and before) time t. U_i, U_f, U_c, U_o denote the weight matrices of different gates for input x_t, and W_i, W_f, W_c, W_o are the weight matrices for hidden state. $h_t.b_i, b_f, b_c, b_o$ are the bias vectors.
For many seq2seq tasks, it is beneficial to have access to both past (left) and future (right) contexts. The solution whose effectiveness has been proven by previous work (Dyer et al., 2015) is bi-directional LSTM (BLSTM). The basic idea is to present each sequence forwards and backwards to two separate hidden states to capture past and future information, respectively. Then the two hidden states are concatenated to form the final output. This step allows the hiden state to capture both past and future information.

CRF A limitation of the single LSTM architecture is that cannot make good use of the output information to get the label. For sequence labeling tasks, it is beneficial to consider the correlations between labels in neighborhoods and jointly decode the best chain of labels for a given input sentence.Therefore, we model label sequence jointly using a conditional random fields (CRFs) with existing LSTM (Lafferty et al., 2001), instead of decoding each label independently. Normally, we use $z = \{z_1, ..., z_n\}$ to represent a generic input sequence where z_i is the input vector of ith word. $y = \{y_1, ..., y_n\}$ represents a generic sequence of labels for z. $\gamma(z)$ denotes the set of possible label sequences for z. The probabilistic model for sequence CRF defines a family of conditional probability $p(y|z; W, b)$ over all possible label sequences y given z with the following form:

$$p(y|z; W, b) = \frac{\prod\limits_{i=1}^{n} \psi_i(y_{i-1}, y_i, z)}{\sum\limits_{y \in \gamma(z)} \prod\limits_{i=1}^{n} (y'_{i-1}, y'_i, z)} \tag{7}$$

The $\gamma_i(y', y, z) = \exp(W^T_{y',y} z_i + b_{y',y})$ are potential function, and $W^T_{y',y}$ and $b_{y',y}$ are the weight vector and bias corresponding to label pair (y', y). In our code, we try to use the negative log likelihood function and the labelwise function to get the loss of the CRF layer. We decode by using the marginal algorithm and viterbi algorithm.
For training example $(x^{(t)}, y^{(t)})$
Log-likehood:

$$L(W, b) = C||w||^2 - \sum_{t=1}^{m} \log(P_w(y^{(t)}|x^{(t)})) \tag{8}$$

Lablewise:

$$R(W) = \sum_{t=1}^{m} \sum_{j=1}^{L} Q(P_w(y^{(t)}|x^{(t)}) - \max_{y_j \neq y_j^t} P_w(y_j|x^{(t)})) \tag{9}$$

2.3 Parameter Initialization

The Parameter Initialization is(best performance):

Table 1. parameters for LSTM-CRF

Optimizer	Embedding	Dropout	Learning rate	Weight dercay	CRF loss
SGD/ADAM	Word2Vec	0.5	0.01	1e-4	lablewise

Randomly select 80% gold standard annotation to be the training data.And the rest to build the dev data set.

3 Single CRF method

We also try to do the single CRF method.

3.1 Proprocessing

Due to the limitations of the existing word segmentation tools in clinical medical text data, a large number of professional words and commonly used medical abbreviations are erroneously segmented, which in turn leads to a large number of boundary error entities in the entity recognition tasks. Therefore, we here cut the medical text based on the Chinese single word directly.

In order to get a better preprocessing result, we train the word segmentation model and part-of-speech(POS) tagging model separately based on the SVM algorithm with the CTB corpus. Also, we use the segmentation model and POS tagging model provided by LTP-Cloud to compare with our own models.

3.2 Feature Extraction

Considered characteristics of the medical text entity, we extract a lot of features, including participle, part of speech, training set of physical dictionary and other characteristics. Finally, we only choose participle and part-of-speech as the valid training features.

For each word, we set length of the window as 5 and then extract features with single feature, 2-gram and 3-gram.

3.3 Model Merging

In view that different training parameters would obtain different models and different models have their own advantages, we train several models using different parameters. Then we merge the models based on the cross-validation results of different categories to get the final model. We choose different model for different category which proform best on this category and merge them into a final result.

3.4 Training Parameters

Here we only give parameters of the best main model, and the differences on parameters of other support models are not obvious except for the Max_iterations.

Table 2. parameters for CRF

Algorithm	L1-penalty	L2-penalty	Max_iterations	Epsilon
passive-aggressive	1e-5	1.0	17	1e-5

4 Result

Here give the final best results of the online judge using LSTM-CRF and Single CRF:

4.1 LSTM-CRF

The best result using LSTM-CRF is:

Table 3. online judge result of LSTM-CRF

	Symptom	Disease	Exam	Treatment	Body	Overall
Relaxed	96.7%	84.4%	95.7%	92.5%	91.4%	93.8%
Strict	94.4%	73.3%	90.9%	71.1%	82.5%	87.3%

4.2 Single CRF

The best result using Single CRF is:

Table 4. online judge result of Single CRF

	Symptom	Disease	Exam	Treatment	Body	Overall
Relaxed	96.3%	86.3%	96.2%	92.8%	91.4%	94.0%
Strict	95.7%	75.6%	93.5%	75.2%	85.9%	89.7%

4.3 Analysis of Result

Compared these two results, we find that disease and treatment don't have a high strict F-measure. Although treatment has a very high relaxed F-measure which means that we have found the right position of treatment without right boundary. The right answer is either too long or too short. As for disease, it seems that the disease entities we find are not much enough. For this issue, result may be better by including extra glossary dictionary. This can be a part of future work.

5 Conclusion

This work presents a system that is completely machinelearning-based; it uses neither a rule-based method nor a postprocessing module. In this named entity recognition task, we achieved a strict F-measure of 0.8974 which ranked sixth. Although we also use LSTM-CRF like other teams, we didn't get a better score. Future work should attempt to adapt the parameter of the LSTM-CRF method in order to get a better F-measure.

6 Acknowledgments

The medical records used in this paper were provided by Jims-Cloud, and CCKS2017 Tasks were supported by the following Grants: Tsinghua Engineering Group, MSRA. Thank you to the organizing committee of CCKS and the annotators of the dataset.

References

1. Schölkopf B, Platt J, Hofmann T. Training Conditional Random Fields for Maximum Labelwise Accuracy[C]// Advances in Neural Information Processing Systems 19, Proceedings of the Twentieth Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December. DBLP, 2006:529-536.
2. Ma X, Hovy E. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF[J]. 2016.
3. Goldberg Y, Levy O. word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method[J]. Eprint Arxiv, 2014.