Report on Clinical Entity Recognition

Mingkun Huang

June 23, 2018

1 Model Architecture

In this section, I'll describe the components of my neural network architecture in fig. 1.

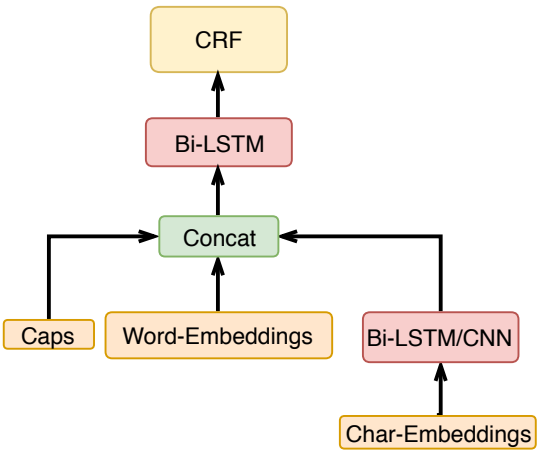

Figure 1: Model architecture

1.1 Input Features

- Word Embeddings: Glove 300d.
- Char Embeddings: random init.
- Capitalism: 4 kinds of types.

Except that, nothing else is used.

1.2 Character-level Representation

Here I use the same structure as described in [1]. But it doesn't work well on this task. Another approach is using Bi-LSTM as feature extracter. Concat the forward and backward hidden vectors, we get the character representation.

1.3 Bi-directional LSTM

Here I use Bi-LSTM as word embeddings featue extracter.

1.4 CRF

After getting the output of word Bi-LSTM, we can use CRF to calculate the likelihood.

2 Experiment

In this section, I provide details about training the neural network. I implement the neural network using the TensorFlow [2] framework [1]. The computations for a single model are run on a GeForce GTX 1080 TI GPU. I do grid search over several hyper parameters, such as optimizer, char embeddings, word embeddings, LSTM layers and CNN or LSTM.

2.1 Grid Search Results

Table 1 shows the results of difference models on development set. Here I only pick several best results. There are over 20 models can get F1 score higher than 84.5 on dev set.

Table 1: Grid search results on development set

Model	Accuracy	Precision	Recall	F1
rmsprop.cap-2.char-100-300.word-300-300.plstm-2	95.03	-	-	85.23
adam.cap-5.char-50-100.word-300-300.plstm-2	95.00	84.90	85.28	85.01
rmsprop.cap-2.char-50-300.word-300-100.plstm-3	94.85	85.47	84.51	84.99
adam.char-20-300.plstm-1	94.85	85.47	84.51	84.99

2.2 Ensemble Training

We select our best 20 models, with F1 score on dev set higher than 84.5, and then load them to a single graph and session. For simplicity, we apply the method described in [3]. The ensemble structure is shown in fig. 2.

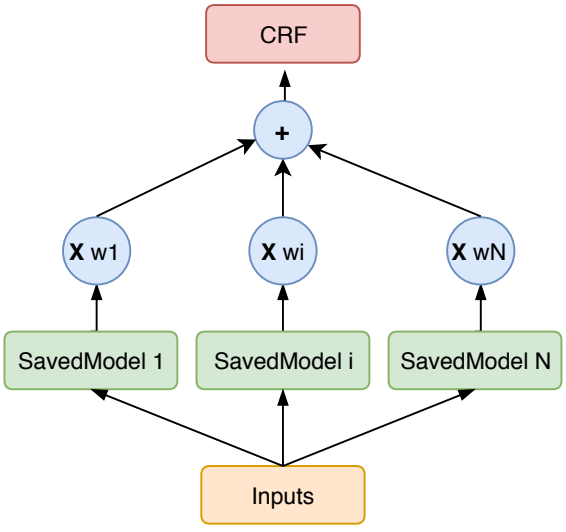

Figure 2: Model architecture: inputs is the same with grid search section, we combine each saved model with differenct structure by multiplying a weight scalar. Each w_i is a probability, and $\sum_i w_i = 1$. And the CRF transition matrix is averaged from saved models.

We do ensemble training using difference combination of models and optimizers. Since those saved model already converged, so we only need one epoch to get the best score on ensemble training. Note that the variables of saved models are not updated, we only update ensemble weights.

[1] https://github.com/HawkAaron/Clinical-Entity-Recognition

We first using 15 best models do ensemble training, and resuts are shown in table 2. We find that sgd can get the best score on dev set. Then we use all 20 best models do ensemble training, and the optimizer is sgd, results are shown in table 3. Finally, the F1 score on test set is shown in table 4.

Table 2: Ensemble training results on development set using 15 best models

Optimizer	Accuracy	Precision	Recall	F1
rmsprop	95.29	86.92	85.18	86.04
adam	95.33	86.95	85.42	86.18
adadelta	95.47	86.87	86.27	86.57
sgd	**95.58**	**87.14**	**86.31**	**86.72**

Table 3: Ensemble training results on **Dev** set using 20 best models

Accuracy	Precision	Recall	F1
95.65	**87.65**	**86.07**	**86.85**

Table 4: Ensemble training results on **Test** set using 20 best models

Accuracy	Precision	Recall	F1
94.72	**83.33**	**85.55**	**84.43**

3 Analysis

3.1 Model Structure

There are many differenct kinds of structure that we can use to do this task. For example, we can use the framework of encoder-decoder with attention to do sequence labelling [4].

3.2 Ensemble Method

Even using the naive ensemble approach described above, we can get significant improvement. There is another method called knowledge distillation [5], can also achieve big improvement over normal trained models.

References

[1] Ma, Xuezhe and Eduard H. Hovy. *End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF*. CoRR abs/1603.01354 (2016): n. pag.

[2] Martn Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Rafal Jozefowicz, Yangqing Jia, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Man, Mike Schuster, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Vigas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. *TensorFlow: Large-scale machine learning on heterogeneous systems*, 2015. Software available from tensorflow.org.

[3] Austin Waters and Yevgen Chebotar. *Distilling Knowledge from Ensembles of Neural Networks for Speech Recognition*, InterSpeech 2016.

[4] Zhu, Su and Kai Yu. Encoder-decoder with focus-mechanism for sequence labelling based spoken language understanding. 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2017): 5675-5679.

[5] Geoffrey Hinton and Oriol Vinyals and Jeffrey Dean. *Distilling the Knowledge in a Neural Network*, NIPS 2015.