Named Entity Recognition

Ramabhadra V
ramavyasa@gmail.com

1 Model 1: Deep sequence tagging model

1.1 Data split

I have split the corpus into 70:10:20 split for training, validation, testing.

1.2 Architecture

Input is a list of lists where each list is a tokenized sentence. Output is a list of lists where each list is set of tags for the corresponding sentence in the input.

There are 3 layers in the model.
a. 1st layer is embedding layer with input 11312(vocabulary) dimensional with output dimension 100.
b. 2nd layer is bidirectional lstm which takes input from embedding layer and with 128 dimensional output. I have used recurrent-dropout 0.1.
c. 3rd layer is the dense layer with softmax activation.

1.3 Results

The following figure plots accuracy of validation and training vs epochs. Best accuracy was after 5th epoch: 97% on validation set.

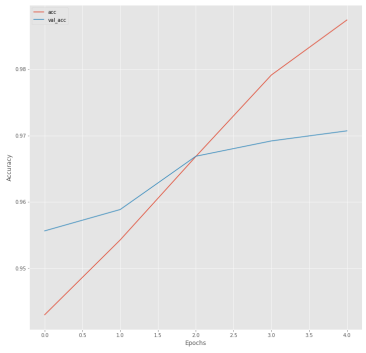

Figure 1: Accuracy vs epochs for deep model.

2 Model 2: CRF

2.1 Data split

I have split the corpus into 80:20 split for training, testing.

2.1.1 Architecture

Each sentence is converted to a list of features where each feature represents a single word in that sentence. This along with list of corresponding tag for each word represents input-output pair.

2.2 features

I have considered these many features for each word.
1. POS tag: I have generated pos tag for each word using NLTK pos-tagger.
2. length of the word.
3. whether the word is a digit.
4. bias of 1.
5. The word itself.
6. last 3 letters and last 2 letters of the word.
5. length, pos-tag, word for previous and next 2 words.

2.3 ablation study

1. All features mentioned above: 0.78
2. with only 1 neighbor: 0.77
3. with no nieghbors: 0.76
3. without pos tag: 0.77
4. without last 3 letters and 2 letters of the word: 0.76
4. wihtout word itself from all neighbors, but all others: 0.59
5. without word only from middle word: 0.73

Hence, the best feature is the word itself.