

CSCI 544 Assignment 4

Mounika Mukkamlla

March 19, 2024

1

For all the three tasks, I initially read train data and saved sentences and tags in a dataframe. Later to that I got word_to_idx and tag_to_idx dictionaries to save indices of words and tags. For unknown words, I assigned index 0. In the next step, I added words_numerical and tags_numerical columns to the dataframe containing numerical feature data of words and tags using word_to_idx and tag_to_idx

2

For BLSTM, in the dataset creation, I used data loaders. and since I am using train data in batches, length of sentences might vary for each training row, hence I padded using pad_collate and utilised collate_fn in DataLoader function.

Next to that, I created a model with embedding layer at first, then bilstm layer with one hidden linear layer and then a classidier with elu activation function.

Hyperparameters I used:

I have used SGD optimiser with learning rate 0.3 and momentum 0.85

I have also used an exponential scheduler with gamma value to be 0.95

For train data, I have used batch size to be 16

Accuracy, precision, recall and F1-score for the dev data are:

accuracy: 93.80 percent; precision: 84.50 percent; recall: 63.50 percent; FB1: 72.51

3

For BLSTM using Glove pre-trained embedded vectors:

All the code is almost same for BLSTM except we use glove pretrained embeddings in the first embedding layer.

Here, case sensitivity is handled through:

In the word embedding matrix, if embedding vector doesn't exist then we convert the word to lower case and find the corresponding embedding vector. If word is not found even after converting to lower case, I returned a np random vector taken from a guassian distribution of embedding_dim size and scale:0.28

Hyperparamters:

I have used SGD optimiser with learning rate 0.3 and momentum 0.85

I have also used an exponential scheduler with gamma value to be 0.95

For train data, I have used batch size to be 16

Accuracy, precision, recall and F1-score for the dev data are:

accuracy: 95.44 percent; precision: 86.84 percent; recall: 74.44 percent; FB1:80.16

4

For BLSTM-CNN, It's the same as task2, except that we use additional Conv1D layer after glove embedding layer.

Hyperparamters:

I have used SGD optimiser with learning rate 0.3 and momentum 0.85

I have also used an exponential scheduler with gamma value to be 0.95

For train data, I have used batch size to be 16

Accuracy, precision, recall and F1-score for the dev data are:

accuracy: 94.52 percent; precision: 61.97 percent; recall: 73.34 percent; FB1: 67.18