Applied Deep Learning
hw2 report

Han Hao Chen, r05922021

November 6, 2016

1 Recursive Neural Network

Rvnn   I use structured nltk tree class to load the parsed pos tree structures, then training on 5000 positive and negative sentences with learning rate 0.01, L2 regularization 0.001, 50 dimensions' random initialized word embedding size and 30 iterations. it can achieve 0.596 accuracy on the CodaLab leaderboard with training accuracy 0.8. However, this model this hard to train and take several days to get the training result. The initialization also is very important. I have run several different and same parameters models on IR Lab server. The result is quit variant even given the same parameters setting. The reason might be the difference of random initialized word vectors. Using the pre-train word embedding vectors may be a possible solution for this model, or using higher dimensions random initialized word vectors.

2 Training Detail

- Given too large L2 regularization will be very hard to promote the accuracy on the training data.

- Even I have a good accuracy on training data, the testing result is still not very good.(ex. 0.7 on training data, only got 5.26 on testing data)

- Loading all logic to the memory will only take few minutes to train the whole model with lot of iterations. However, this is not suitable for large data (maximum 500 pos tree for mac pro 8G)

- Logic of each tree will keep in the tensorflow graph, so I need to exit the graph and open another one, then restore the model variables(after loading 50 structured trees)

References

[1] https://github.com/yihui-he/recursive-neural-network