Recursive Neural Network for Sentiment Analysis

National Taiwan University Electrical Engineering Chia Hsuan-Li

Data

The input data are sentences parsed by The Stanford Parser.

Model

Wordembedding：Pretrain all the input strings by tensorflow word2vec
https://github.com/tensorflow/tensorflow/tree/master/tensorflow/models/embedding

Recursive Neural Network：

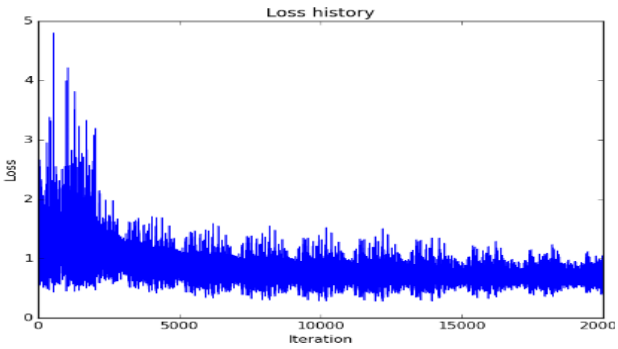
When two words merge, we implement a NN and tie weights on every node. Because we don't have label on every word, so we only predict the result on root. (Predict the sentiment on one sentence 0/1, negative or positive)

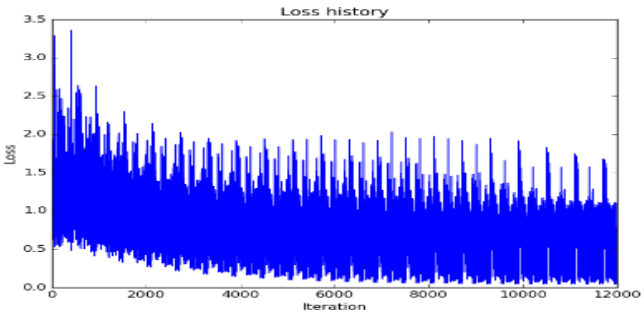
Current hyper-parameters and performance

Word_embed_size	max_epochs	fixed_tree	acc
200	50	no	0.526
40	20	no	0.5
40	10	yes	0.512

The uploaded code provides perfect parser for pure-binary tree. But there are only 640 sentences and there are 5000 training sentences in total, making the training difficult.
I didn't complete the parser for multiple children tree, so i use a fixed-form instead.
Merge every two-word in the sentence backwards. There are 1000 sentences in testing data. But there are only probably 150 sentences in them. The remaining sentences are predicted randomly. Making the accuracy varies and hard to evaluate the performance of the models.
So I turn to see the loss history of them.

rnn_embed=40_epoch=20 *rnn_embed=40_ epoch=10_addfixedtree*





As expected the loss for the one with more training data converges earlier. But the performance isn't better.

Wordembedding dimension influence：

In conclusion, I believe the wordembedding dimension has to reach certain size to capture the word meaning and also the W paprameter needs to be bigger to be able to produce complicated transformation on each node.

TO DO : I plan to complete the parser or find a way to transform the tree into pure binary form.