Sravani Konujula
16230172
ID :- 23

Deep Learning
LAB Assignment-3

Introduction

The main aim is to implement the text classification using CNN model, RNN model, LSTM model with new data set and compared results obtained.

Objectives
To compare the accuracy results between RNN, CNN, LSTM.

Approaches/Methods
Upload the dataset from the regret and later load it.
Later train it and evaluate it.
By setting the parameters and hiding layers run the code to get wanted results.

Workflow

- Required libraries are to be imported such as Numpy, Pandas, Keras, text CNN, RNN, LSTM
- Parameters are considered for Data, Model and Training
- Data to be used for pre-processing
- Load the wanted data and build vocabulary then shuffle randomly
- Splitting is done as the test and train data
- Cross validation must be done after training the data
- Train the procedure
- Placeholders are created.
- Summarize the loss and accuracy
- Evaluate parameters
- Observe the results

Datasets
The data set considered is regret.txt

Parameters

For CNN

```
Parameters:
ALLOW_SOFT_PLACEMENT=True
BATCH_SIZE=64
CHECKPOINT_EVERY=100
DEV_SAMPLE_PERCENTAGE=0.01
DROPOUT_KEEP_PROB=0.5
EMBEDDING_DIM=128
EVALUATE_EVERY=100
FILTER_SIZES=3,4,5
L2_REG_LAMBDA=0.0
LOG_DEVICE_PLACEMENT=False
```

For RNN

```
# Parameters
learning_rate = 0.001
training_iters = 1000
display_step = 500
n_input = 3
```

For LSTM

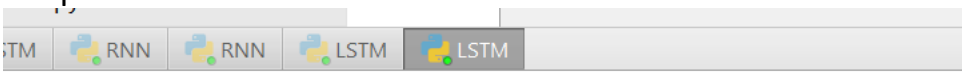
```
# Parameters
learning_rate = 0.001
training_iters = 1000
display_step = 500
n_input = 5
```

Configuration

Pycharm
Python: 2.7.13
Tensor Flow
Keras

Evaluation & Discussion

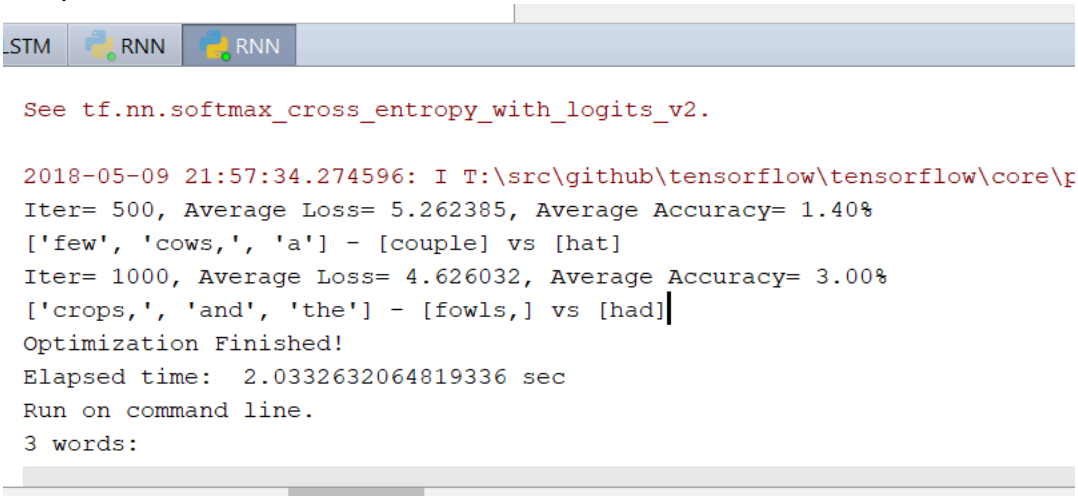
Output for LSTM



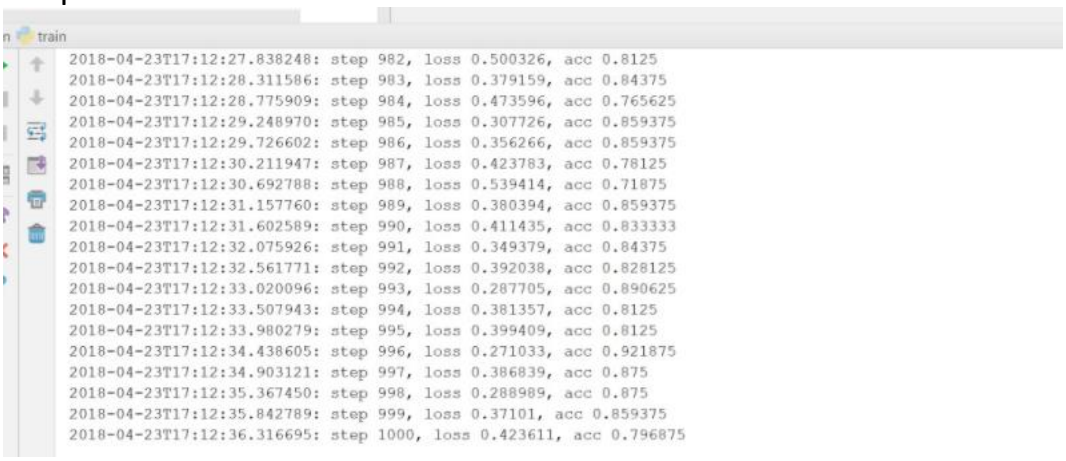

```
See tf.nn.softmax_cross_entropy_with_logits_v2.

2018-05-09 22:03:00.246904: I T:\src\github\tensorflow\tensorflow
Iter= 500, Average Loss= 4.986780, Average Accuracy= 6.40%
['marrying.', 'She', 'had', 'never', 'been'] - [in] vs [a]
Iter= 1000, Average Loss= 3.444704, Average Accuracy= 15.20%
['yet', 'lived', 'to', 'regret', 'it.'] - [So] vs [in]
Optimization Finished!
Elapsed time:  2.84920072555542 sec
Run on command line.
5 words:
```

Output for RNN



Output for CNN



Conclusion

From the above screenshots we can observe that CNN performance is too good compared than the other too. So, CNN is best and Later LSTM compared to RNN. The ranking will be CNN, LSTM, RNN.