CSEE5590/490:

Python and Deep Learning Programming

(2018 Fall)

*Python Lab Assignment 4*

Submitted On:

04 December, 2018

Submitted By:

Name: Farid Uddin Ahmed

Class ID: 02

Name: Zarin Tasnim Sandhie

Class ID: 26



LIST OF THE DOCUMENTATIONS:

1. Author
2. Objective
3. Features
4. Configuration
5. Input/ Output Screenshots
6. Implementation & Code Snippet
7. Evaluation & Explanation
8. Conclusion
9. Links
10. References

AUTHORS

This report contains all the documents for the Lab Assignment #3. The assignment was done by Farid Uddin Ahmed (Class ID 2) and Zarin Tasnim Sandhie (Class ID 26), both are graduate student majoring in Electrical Engineering Department at University of Missouri Kansas City (UMKC).

OBJECTIVE

In the last four weeks of Python/deep learning class, we got to work with different kinds of text classification and image classification. We used LSTM, RNN and CNN for both of the classification. The definition and working principle of the following topics were taught in the class:

* Neural Networks
* TensorFlow
* Feedforward Neural Network (FNN)
* Recurrent Neural Network (RNN)
* Long Short Term Memory (LSTM)
* Text Classification using RNN and LSTM
* Convolutional Neural Networks (CNN)
* Image classification using CNN

This assignment includes all of the things mentioned above.

* The dataset chosen for problem 1, 2 and 3 is twitter airlines dataset. This is a text dataset, as a result, text classification is used. Sentiment analysis is done for these problems.
* For problem 1, text classification is done with CNN model with the above mentioned dataset.
* For problem 2, text classification is done with RNN and LSTM model with the same dataset.
* In problem 3, the results using different model is compared on the basis of accuracy and loss.
* Problem 4 works with image classification. For this problem, we have used Food 101 dataset.

FEATURES

The features of all the problems are discussed below:

Problem 1:

Implement the text classification with CNN model, with a new dataset which is not used in the class.

Problem 2:

Implement the text classification with RNN/LSTM model, with a new dataset which is not used in the class.

Problem 3:

Compare the results of CNN and RNN/LSTM models, for the text classification (same dataset for two models to compare) and describe, which model is best for the text classification based on your results

Problem 4:

Implement the image classification with CNN model, with a new dataset which is not used in the class (E.g. CIFAR 10 dataset)

CONFIGURATION

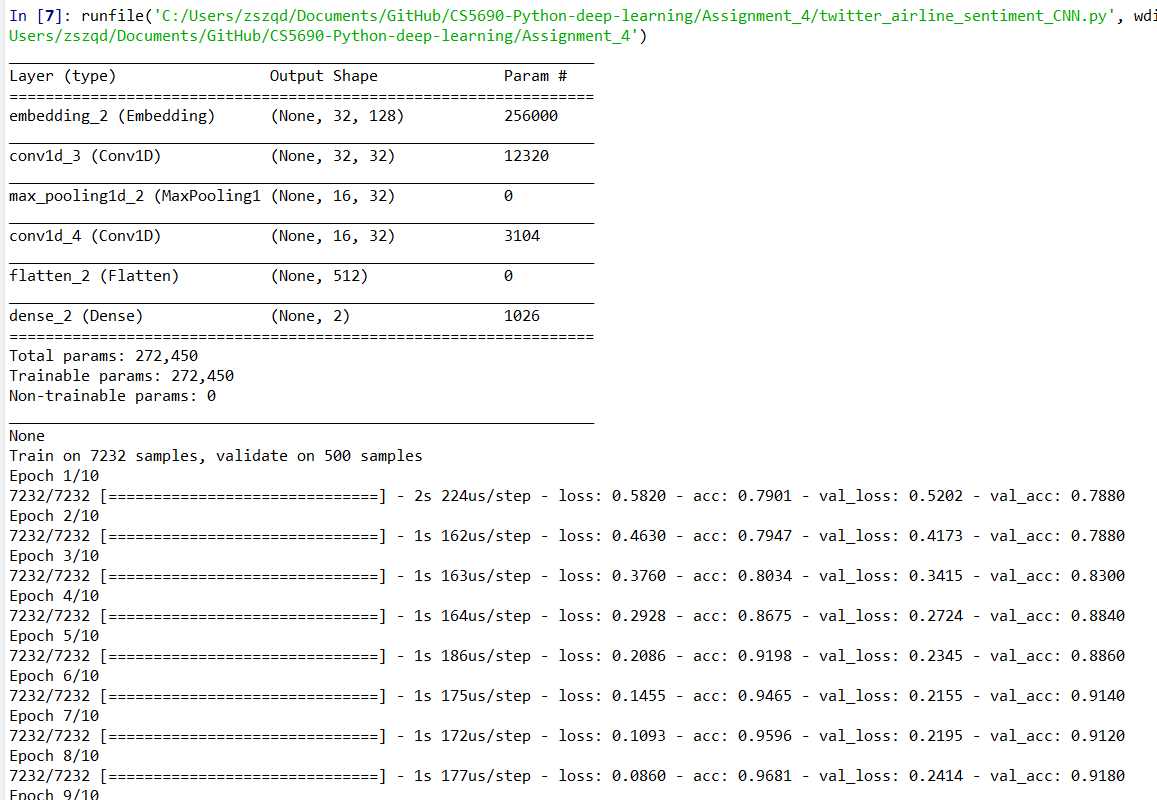
For executing the assignment, coding was done with Python software version 3.6. The simulation was in software: Anaconda (Spider).

INPUT/OUTPUT SCREENSHOTS

Problem 1:

* The “Twitter airlines” datasets in the keras library is taken as input dataset.
* CNN model is used in this problem.
* When simulated, the output plot shows the training and validation loss with the change of epochs.
* Training and validation accuracy graph is also shown.

Output:



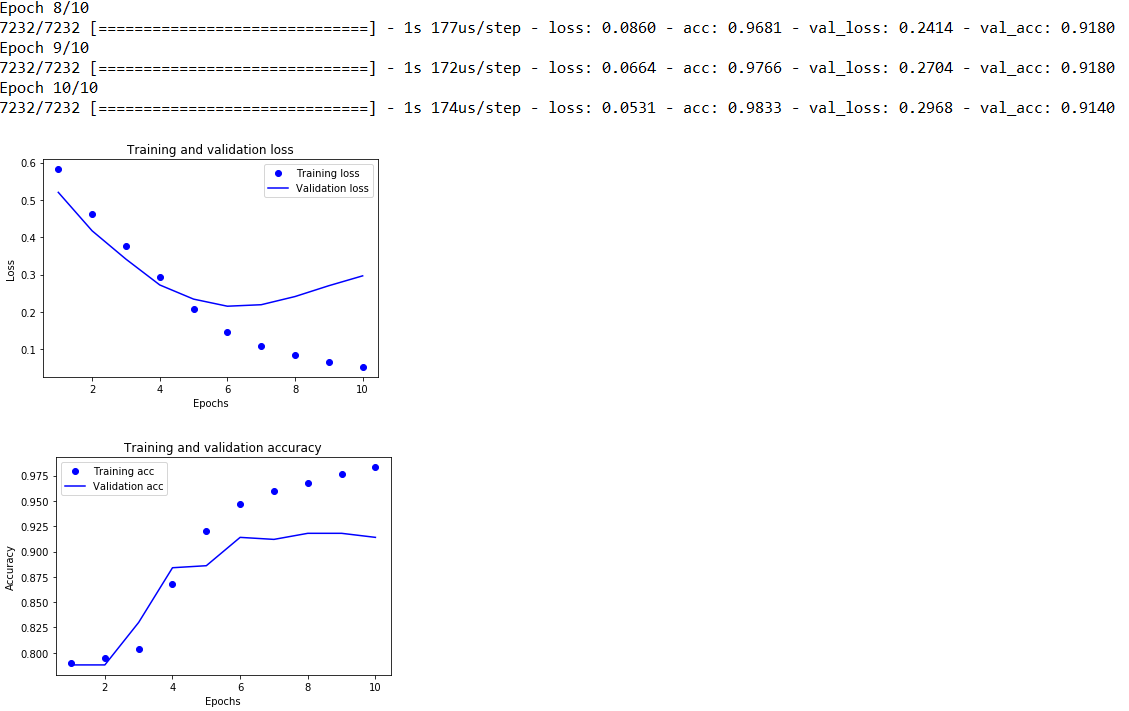
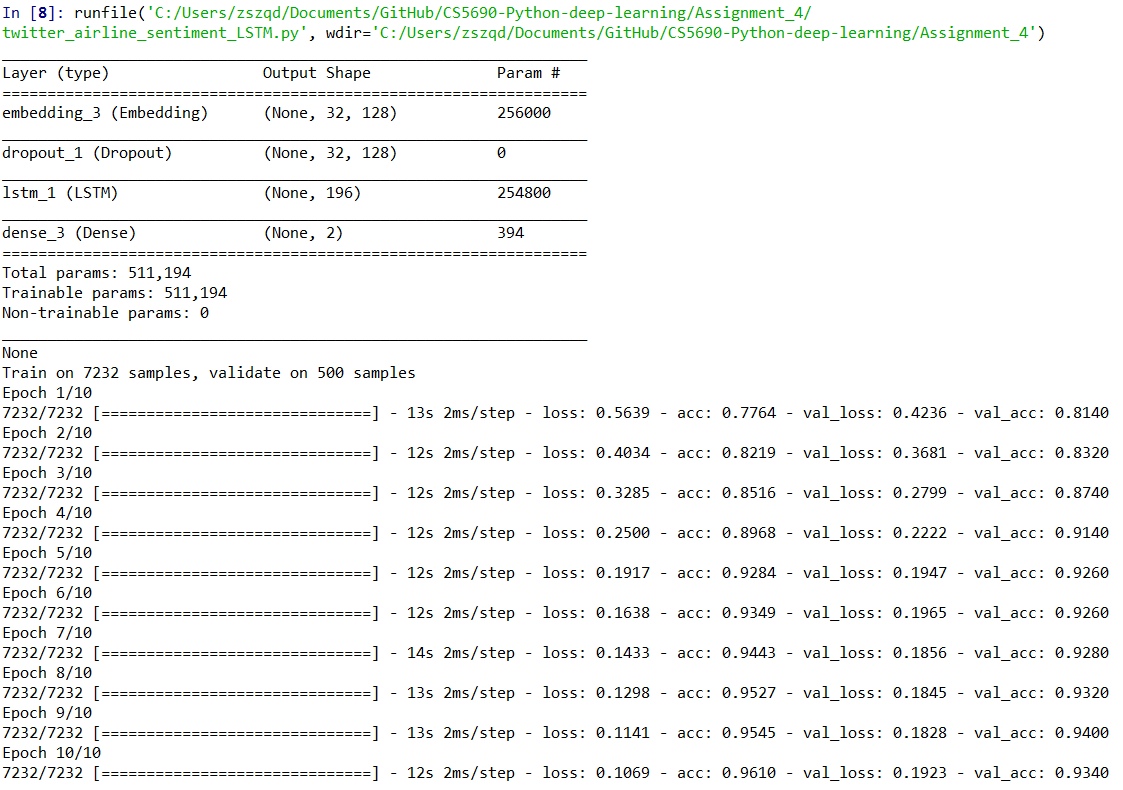


Figure 1: The model specification and training and validation loss and accuracy graph for CNN model

Problem 2:

* The “Twitter airlines” datasets in the keras library is taken as input dataset.
* Both LSTM and RNN models are used in this problem.
* When simulated, the output plot shows the training and validation loss with the change of epochs.
* Training and validation accuracy graph is also shown.

Output for LSTM model:



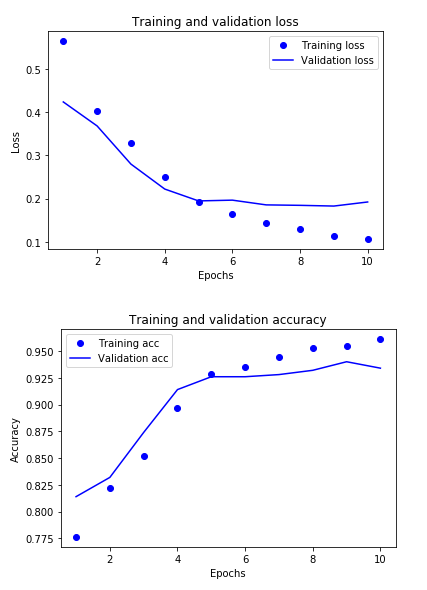
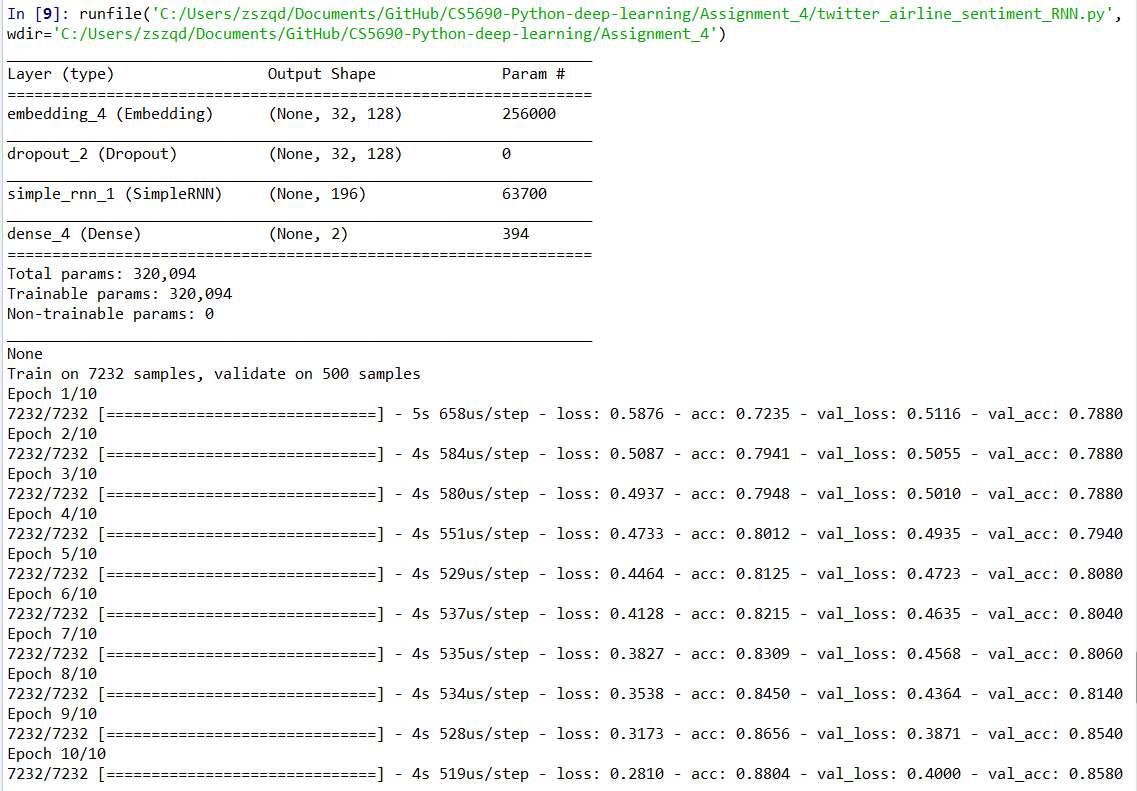


Figure 2: The model specification and training and validation loss and accuracy graph for LSTM model

Output for RNN model:



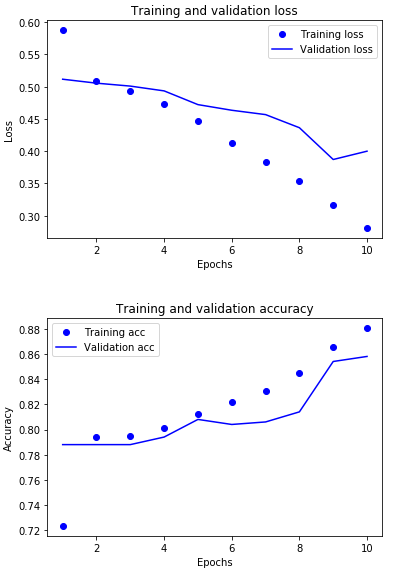


Figure 3: The model specification and training and validation loss and accuracy graph for RNN model

Problem 3:

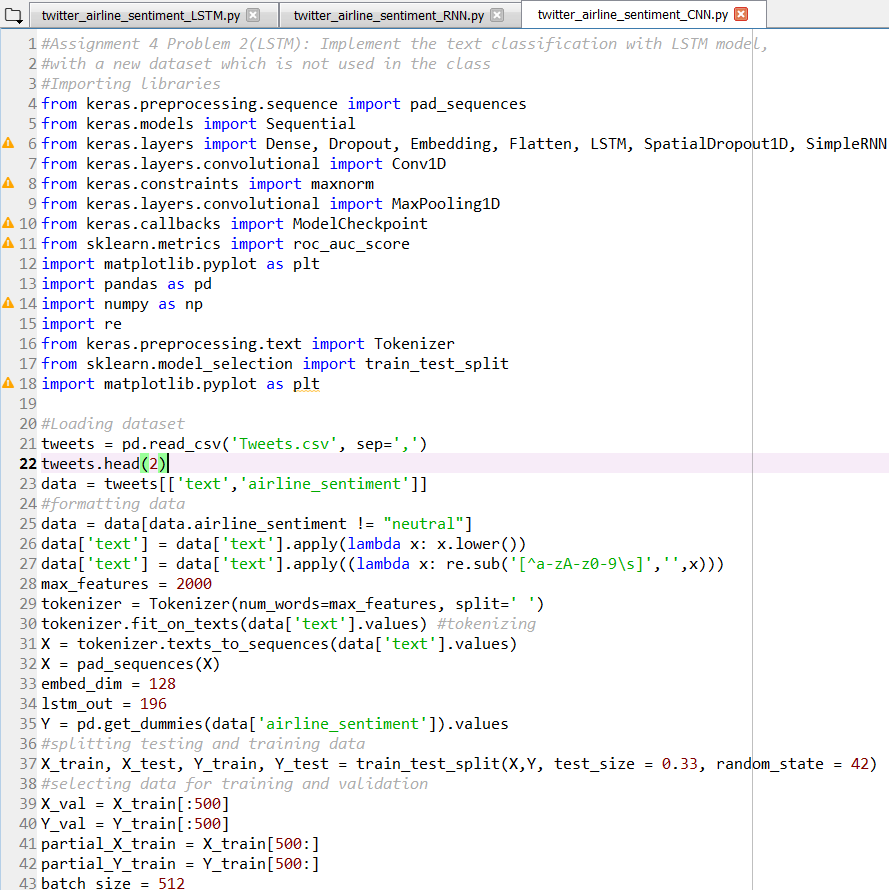
The comparison between all the models is done. Among all the models, CNN is best in terms of training accuracy. But LSTM is best in terms of validation accuracy. Again, CNN gives better result for training loss. And LSTM gives better result in terms of validation loss. RNN gives the worse result for all the cases.

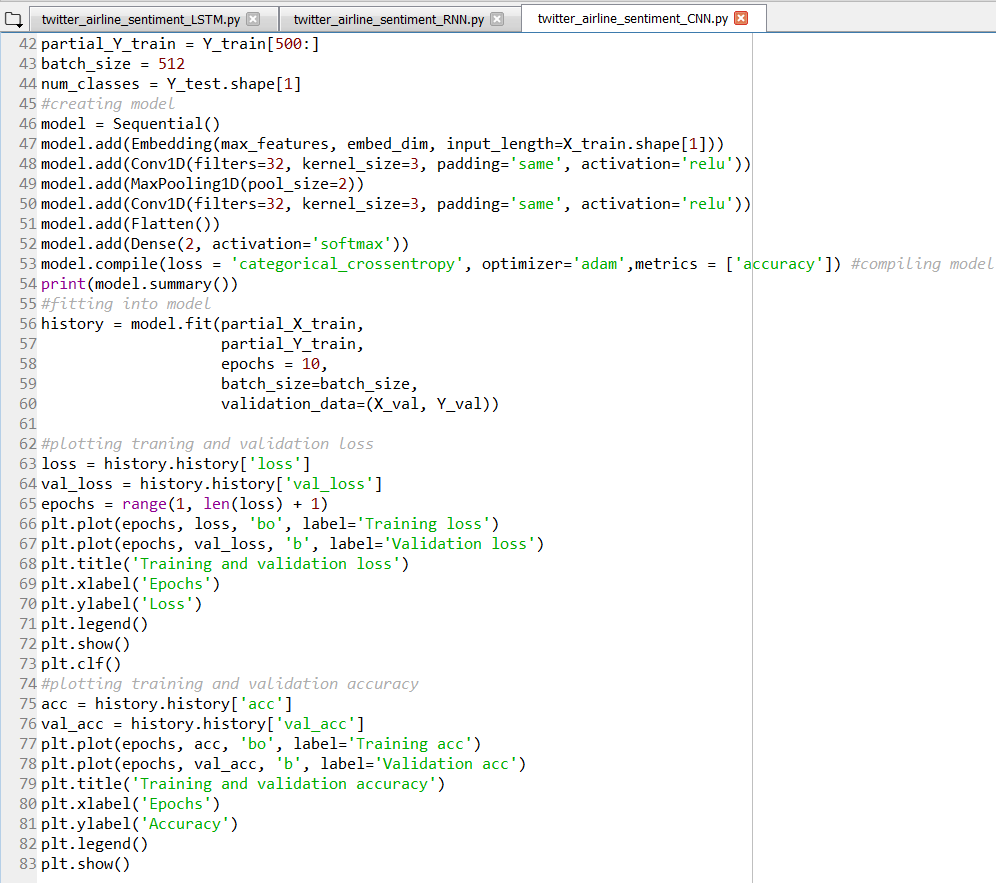
IMPLEMENTATION & CODE SNIPPET

Problem 1:

* Libraries are imported.
* The code takes “twitter airline” dataset as input dataset.
* The dataset is split into testing and training dataset.
* A model is created with sequential command.
* CNN is applied on that model.
* All the parameters are given accordingly.
* After, compilation and fitting of the model, the plot is showed.

Code:

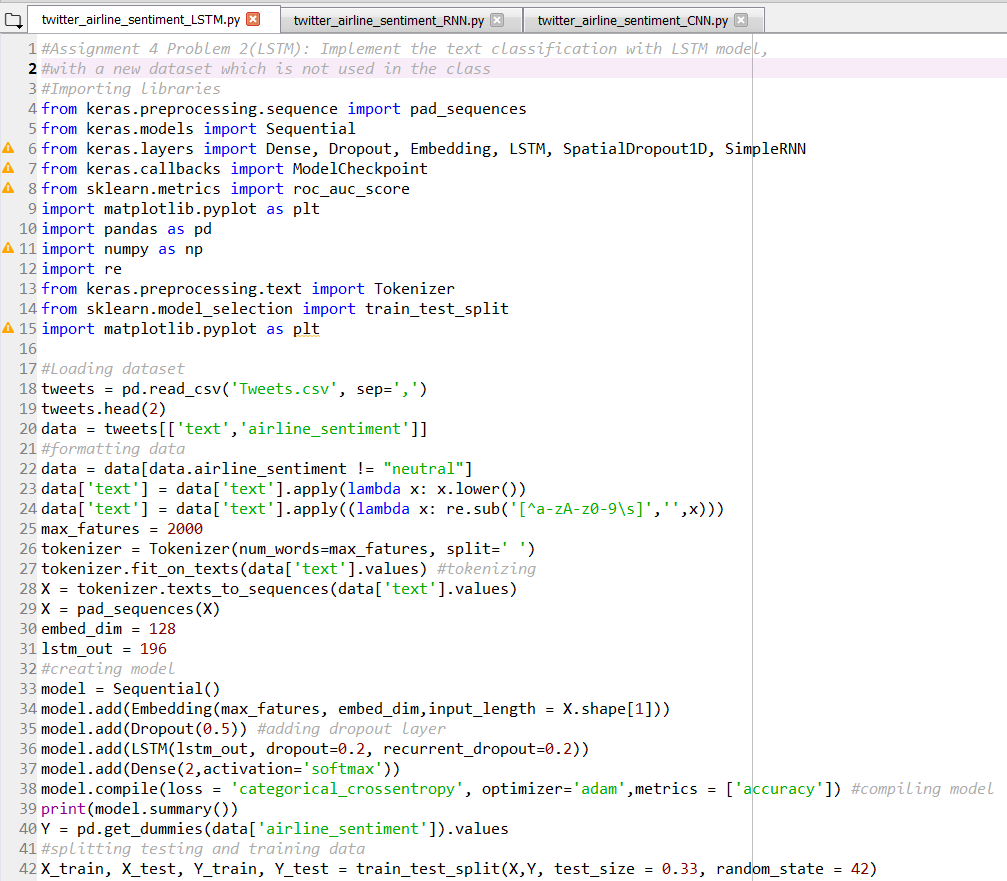


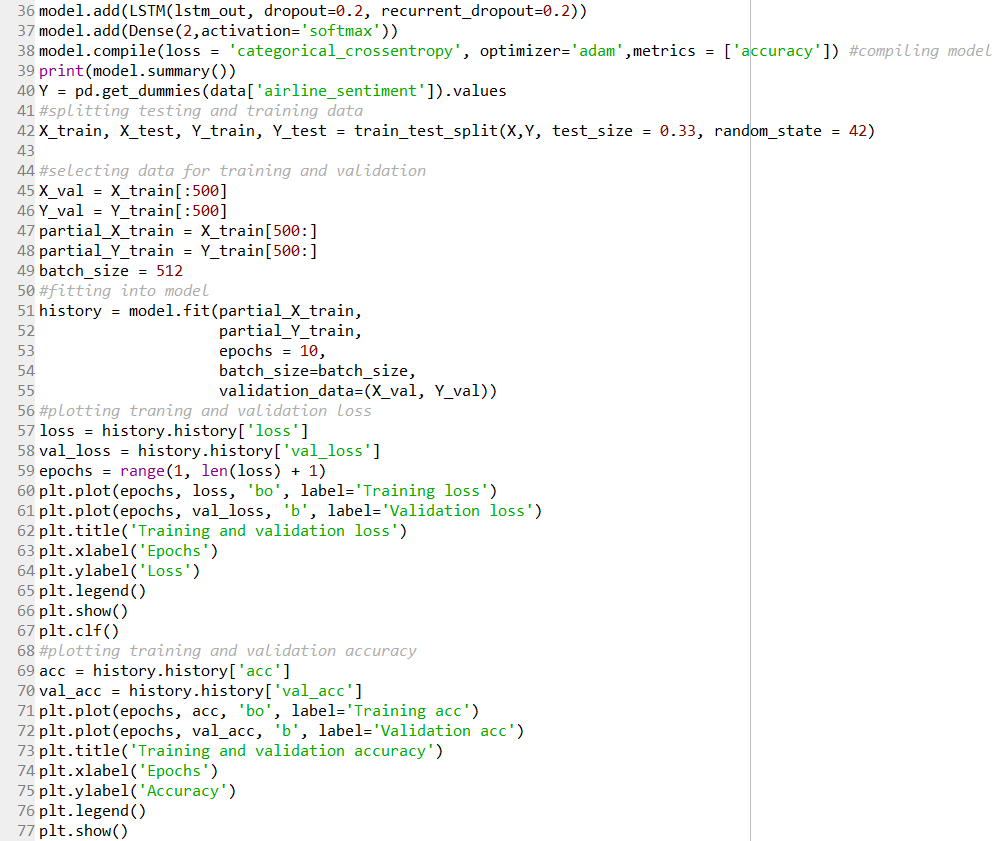


Problem 2:

* Libraries are imported.
* The code takes “twitter airline” dataset as input dataset.
* The dataset is split into testing and training dataset.
* A model is created with sequential command.
* LSTM and RNN is applied on that model.
* All the parameters are given accordingly.
* After, compilation and fitting of the model, the plot is showed.

Code (LSTM):





EVALUATION & EXPLANATION:

Problem 1:

Effect of changing learning rate:

When the learning rate is increased (from 0.001 to 0.01), the accuracy increases. Lower learning rate or higher learning ratio means fast learning. For that case, accuracy becomes worse. Higher learning rate or lower learning ratio means accurate learning. As a result, accuracy gets better.

Effect of changing batch size:

If batch size is increased, simulation gets fast. But the loss increases. When batch size is decreased, simulation gets slow but the loss decreases. The reason behind this is that, with increased batch size, the number of examples used in a batch increases, so the timing is decreased and the result may be affected.

Effect of changing optimizer:

We tried for two types of optimizer: RMSprop and Adam. The results were better for RMSprop type of optimizer.

Effect of changing activation function:

For the problem, two different types of activation functions were used: Relu and tanh. The accuracy and loss were better with “relu” type of activation function.

Problem 2:

Effect of changing batch size:

If batch size is increased, simulation gets fast and if it is decreased, simulation gets slow. In this logistic type of regression, with the increase of batch size, the quality of accuracy gets better.

Effect of changing optimizer:

The result is similar to the first problem. The results were better for RMSprop type of optimizer.

Effect of changing activation function:

This case was also similar to the previous problem. Among the two different types of activation functions, Relu and tanh, the accuracy and loss were better with “relu” type of activation function.

CONCLUSION

All the required problems were solved successfully during this assignment. During this assignment, we learned the use and advantages of two different kinds of regression: Linear and Logistic. We also learned how the different parameters affect the accuracy and loss of the process. And we observed the results with both matlabplot and TensorBoard.

LINKS

Video Link:

<https://www.youtube.com/watch?v=ALChDLsXPX0&feature=youtu.be>

Github Link:

https://github.com/Sandhie177/CS5690-Python-deep-learning/tree/master/Assignment\_3

REFERENCES

1. <https://www.linkedin.com/pulse/list-useful-links-videos-slides-articles-deep-farshid-pirahansiah?articleId=6274891035365732352#comments-6274891035365732352&trk=prof-post>
2. <https://www.kaggle.com/saurabh00007/diabetescsv>
3. <http://www.machineintellegence.com/logistic-regression-in-keras/>