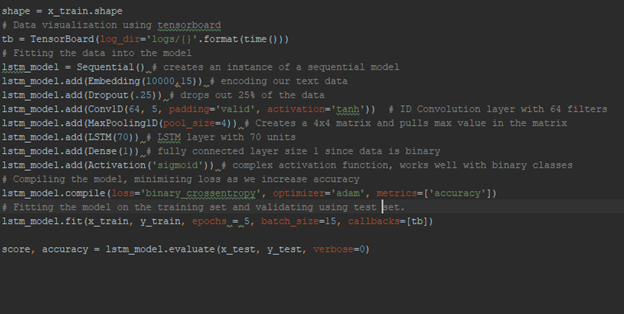
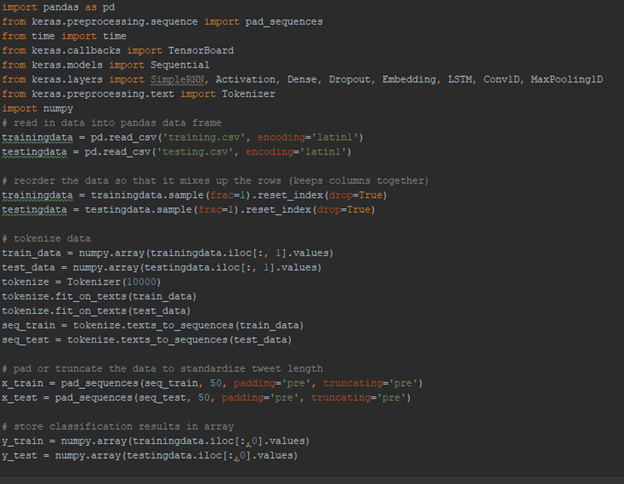
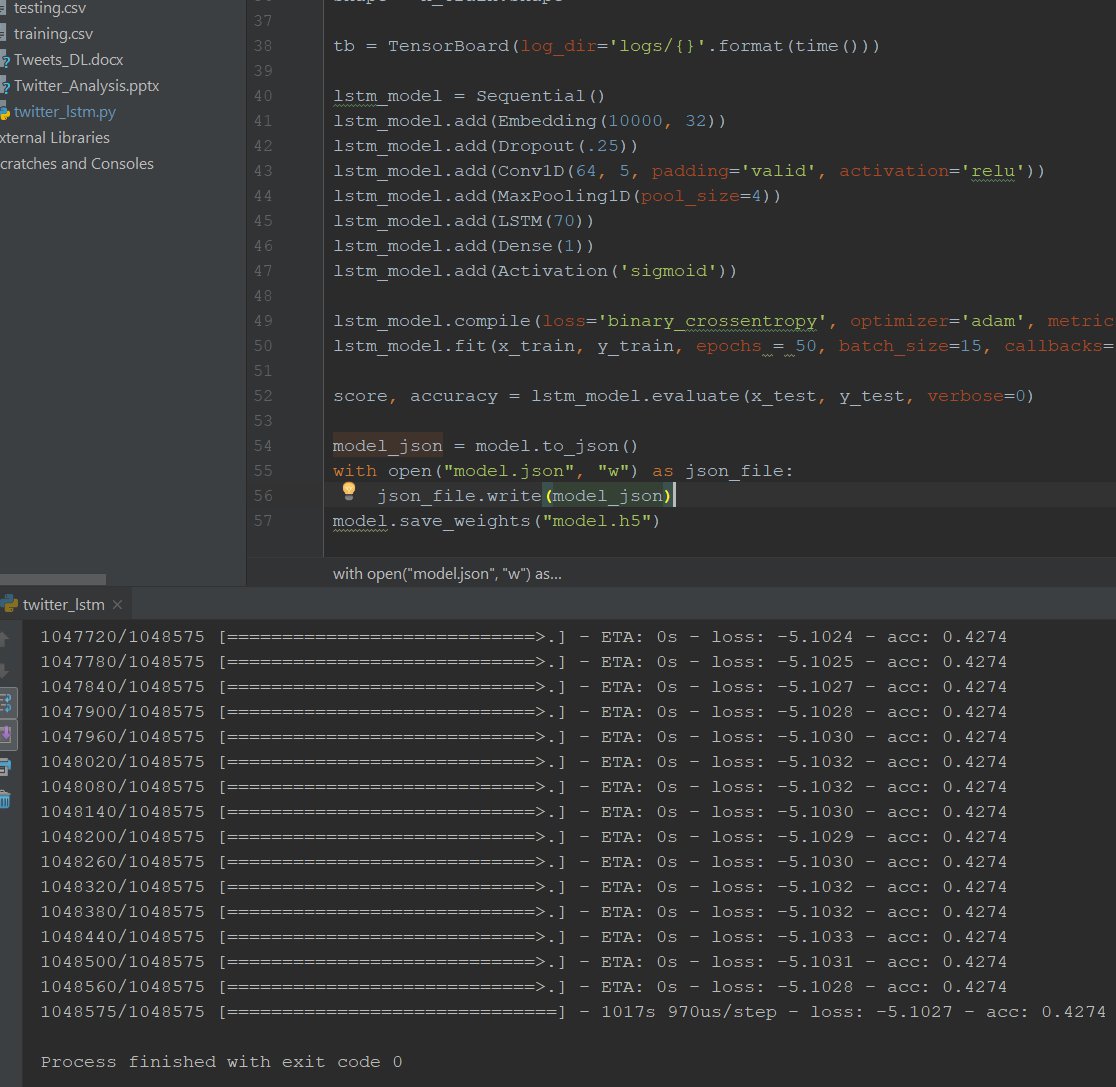
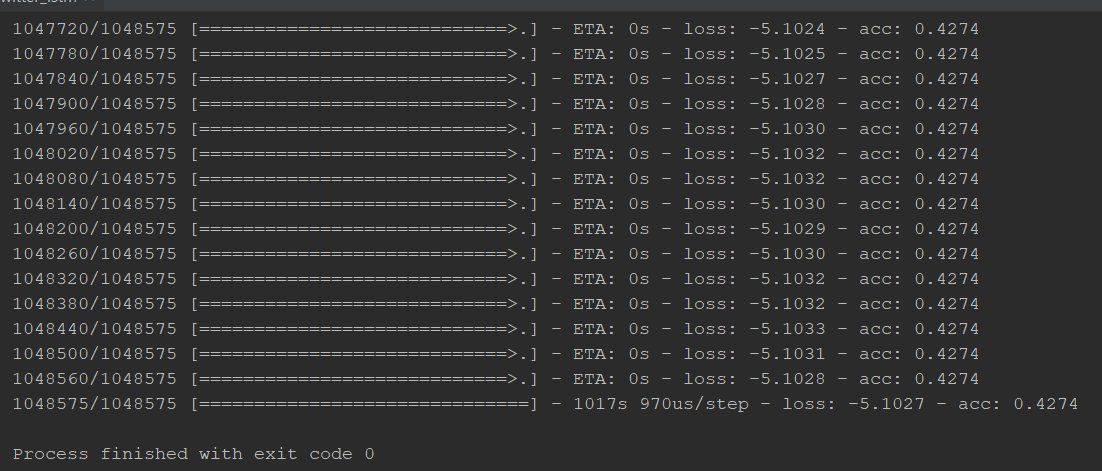
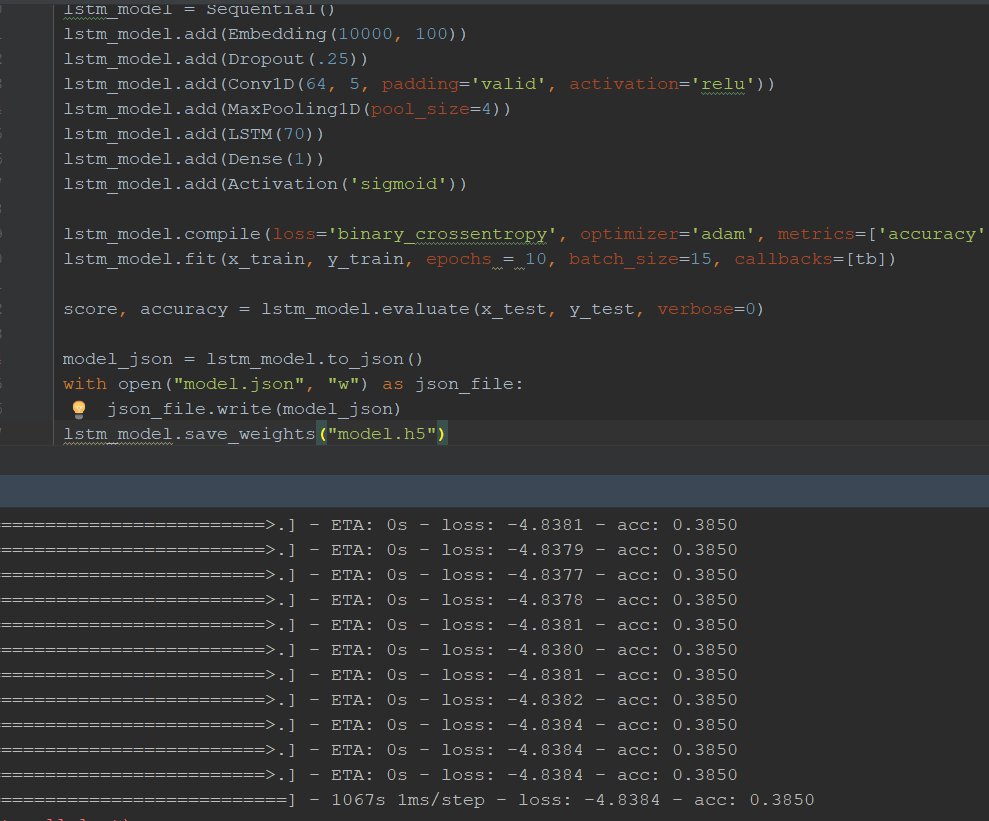
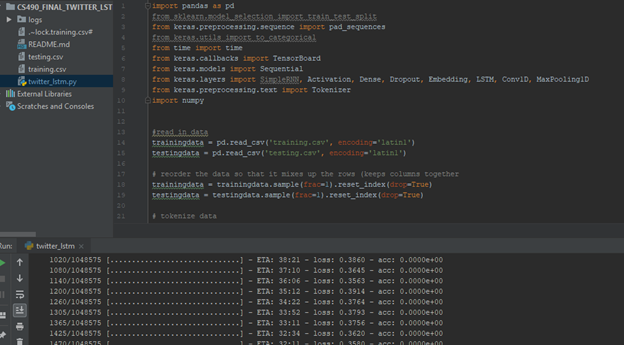
CSEE 5590 APS-Python Programming  
Fall 2018  
Assignment: Python Lab4   
Presented By: Piatt Alexandria, Larios Alexander, Elisha Sigei  
Class IDs:23,15,29  
Final Project  
Submission Date:12/10/2018  
  
Department of Computer Science and Electrical Engineering

**Table of Contents**  
Introduction……………………………………………………….. 3  
Objectives…………………………………………………………. 4  
Approaches/Methods……………………………………………..5  
Input/Output Screenshots……………………………………….. 8  
Datasets…………………………………………………………... 11  
Integrated Development Environment (IDE)............................ 12  
Conclusion………………………………………………………... 13  
 

**INTRODUCTION**  
  
This project focused on employing deep learning techniques we acquired through the semester on processing and analyzing Twitter sentiments. Twitter is one of the top social networks with over 336 million active users monthly. Twitter has become a major platform for marketing, politics and even exchange of ideas like disease tracking. The main reason why we chose twitter is that posts are more concise and accurate due to the limitation on the number of characters that one can type per tweet. Twitter allows approximately 240 characters which on average equals to about 40 words per tweet.

**OBJECTIVES**  
The main objective of this project is to use Machine Learning techniques to process and determine the sentiments of various tweets. Our approach involves using the LSTM (Long Short-Term Memory) model since LSTM model works best for data classification that has time sensitivity. Another reason for choosing LSTM is that, LSTM model can remember previous data and make a forecast based on it. Unlike RNN, LSTM has a wider range of memory span. In this project, we focused on general sentiments analysis. Our inspiration was drawn from disease tracking use case. In this scenario, it needed a very time sensitive model as it helps researchers to get the most relevant and meaningful information. Our future scope for this project is to build a model that uses real-time stream of data.  
  
  
  
  
  
  
  
  
  
  
  
  
**APPROACHES AND METHODS**  
**Twitter Sentiments Analysis Using LSTM**  
Our main goal is to build an LSTM sequential model. In this task, we used Sentiment140 data set. Sentiment140 dataset is a common and massive twitter data. The dataset includes over one million tweets in the training set, this dataset has several features but for the sake of our project, we extracted texts and sentiments. The tweets are pre-cleaned hence, easy to use. The task for this project involved pre-processing the data and fitting it into the sequential model.  
Pandas, keras and NumPy libraries aided our data pre-processing. We used Pandas to read our input data into a data frame, and reshuffling of rows since data was ordered by sentiment. Kera’s library was used to tokenize tweets, then each tweet padded and truncated to the same length and finally used NumPy to store our data into a NumPy array.  
After data was processed, we created an instance of the sequential model. Then on our model, we added an embedding layer to encode the text data into a language that the neural network understands. We then added a dropout layer for data regularization. We then added a 1-dimensional convolutional neural net layer to help condense the sequence for LSTM layer processing. We used 64 filters in the Conv1D layer and rectifier activation – this is a complex function and it fits well with a deep neural network. We also added a MaxPooling1D layer with a pool size of 4 – this layer helps to keep the relationship between data. Our model is built up with many layers. We then added a LSTM layer with 70 units and a fully connected dense layer that further flattens the data. We used another complex activation function on our last layer. We used a sigmoid activation function since it works best for binary classification data. Tweets sentiments are characterized as either positive or negative.  
On the final part, we compiled and took the loss and accuracy metric. We used binary cross-entropy – common loss function allows for multiple labels to be applied to one input. We also used ‘adam’ optimizer since it is more flexible and worked best for our dataset.  
**Below is the code:**  
   
   
  
**INPUT/OUTPUT SCREENS**  
  
  
  
  
With 5 epochs we got an accuracy of 42.74%.  
  
  
  
  
  
Tried Running it longer for 10 epochs accuracy went down?

With 10 epochs the accuracy went down to 38.50 percent. Some of the challenges we encountered running this project includes time efficiency. Our project takes large amount of time running since we have a lot of input data and processing is slow.  
  
  
  
  
  
  
  
  
**DATASETS**  
In this task, we used Sentiment140 data set. Sentiment140 dataset is a common and massive twitter data. This dataset has many tweet features, but we extracted only texts and sentiments.  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
**INTEGRATED DEVELOPMENT ENVIRONMENT**  
PyCharm was us to execute this lab, python version 3.6.4  
 

**CONCLUSION:**  
In this project, we focused on applications of neural networks in sentiments analysis. This project focused on using LSTM (Long Short-term Memory) model for tweets sentiments analysis. Since this is a deep neural network, we used complex activation functions namely rectifier and sigmoid as this offers a better fit of the data. Long short-term memory network layer was added on CNN layer as this helps in remembering the inputs. As seen from the output screens above, with 5 epochs we got an accuracy of 42.74 percent while with 10 epochs we got 38.50 percent accuracy. One of the problems we experienced is that running this large amount of data took too much time with just 5 and 10 epochs. Our future scope of this project type will involve time optimization, we will be working on accuracy improvement and processing time reduction.