

**Python and Deep Learning**

**# Lab 4 Assignment**

**Team Members:**

Pragathi Thammaneni

Sridevi Mallipudi

LAB -4 Assignment

**Introduction:**

The key objective of this assignment is to focus on text classification with CNN model, RNN/LSTM models and image classification using CNN model tensor flow deep learning library**.**

**Objectives:**

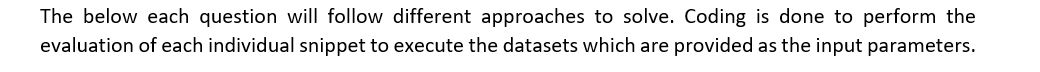
To code for the 4 questions by implementing the below concepts**.**

* Text classification - CNN model
* Text classification – RNN /LSTM model
* Hyperparameters of CNN
* Hyperparameters of RNN / LSTM model
* Comparison between the results of CNN vs RNN / LSTM
* Image Classification -CNN model

**Approaches /Methods:**

Using Python 3.6, PyCharm (Community edition)

**Workflow &Datasets/Parameters and Evaluation:**

****

**A screenshot of a cell phone

Description generated with high confidence**

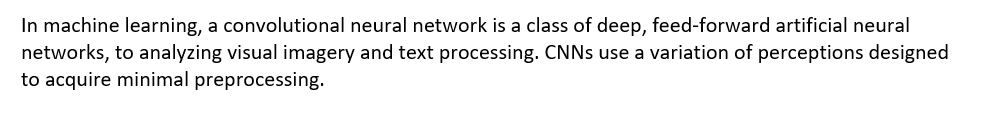
**Question 1:**

**A close up of a device

Description generated with high confidence**

**Solution:**

**CNN Text Classification:**



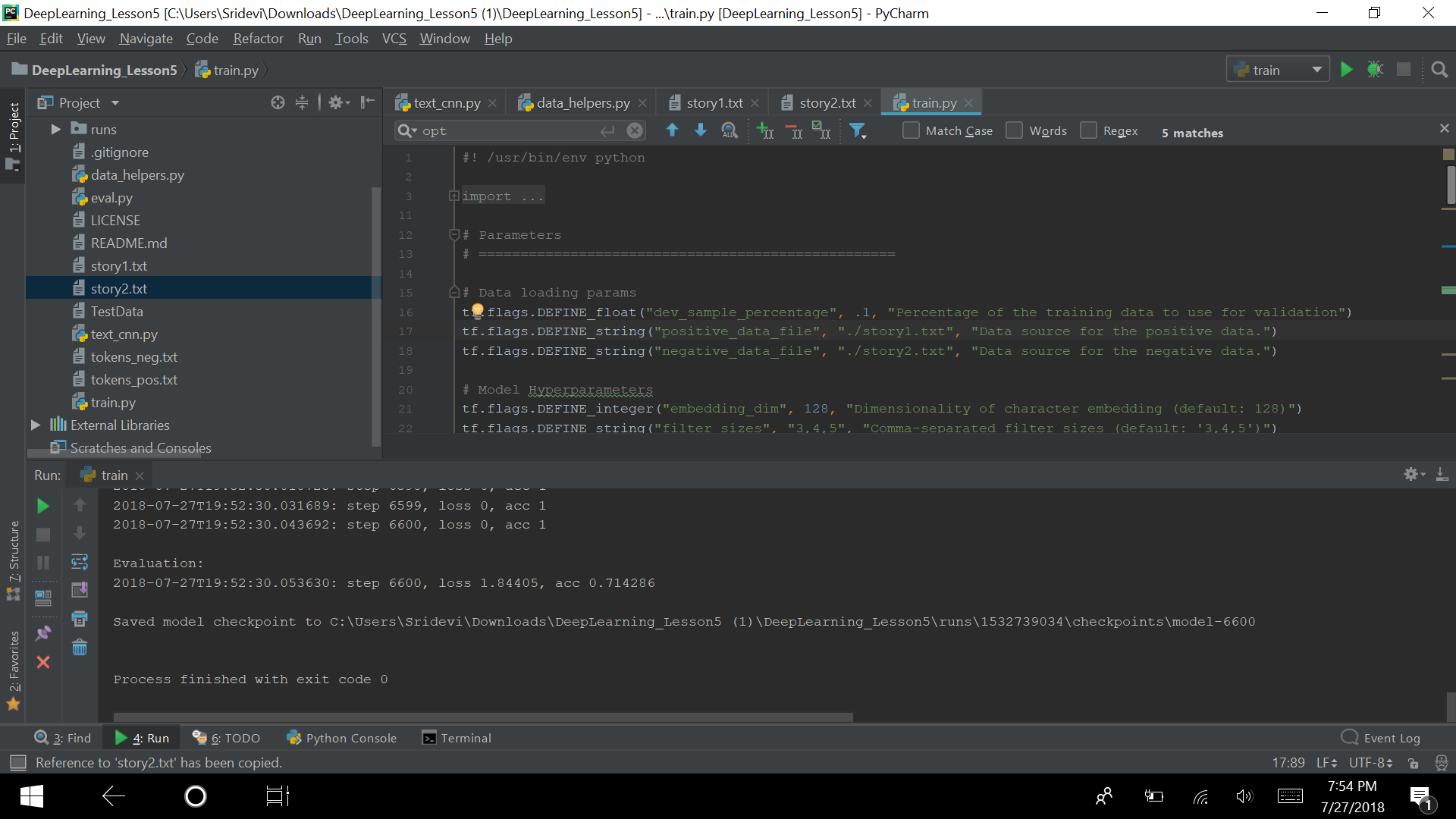
This snippet code provides the above implementation

**Data Set:**

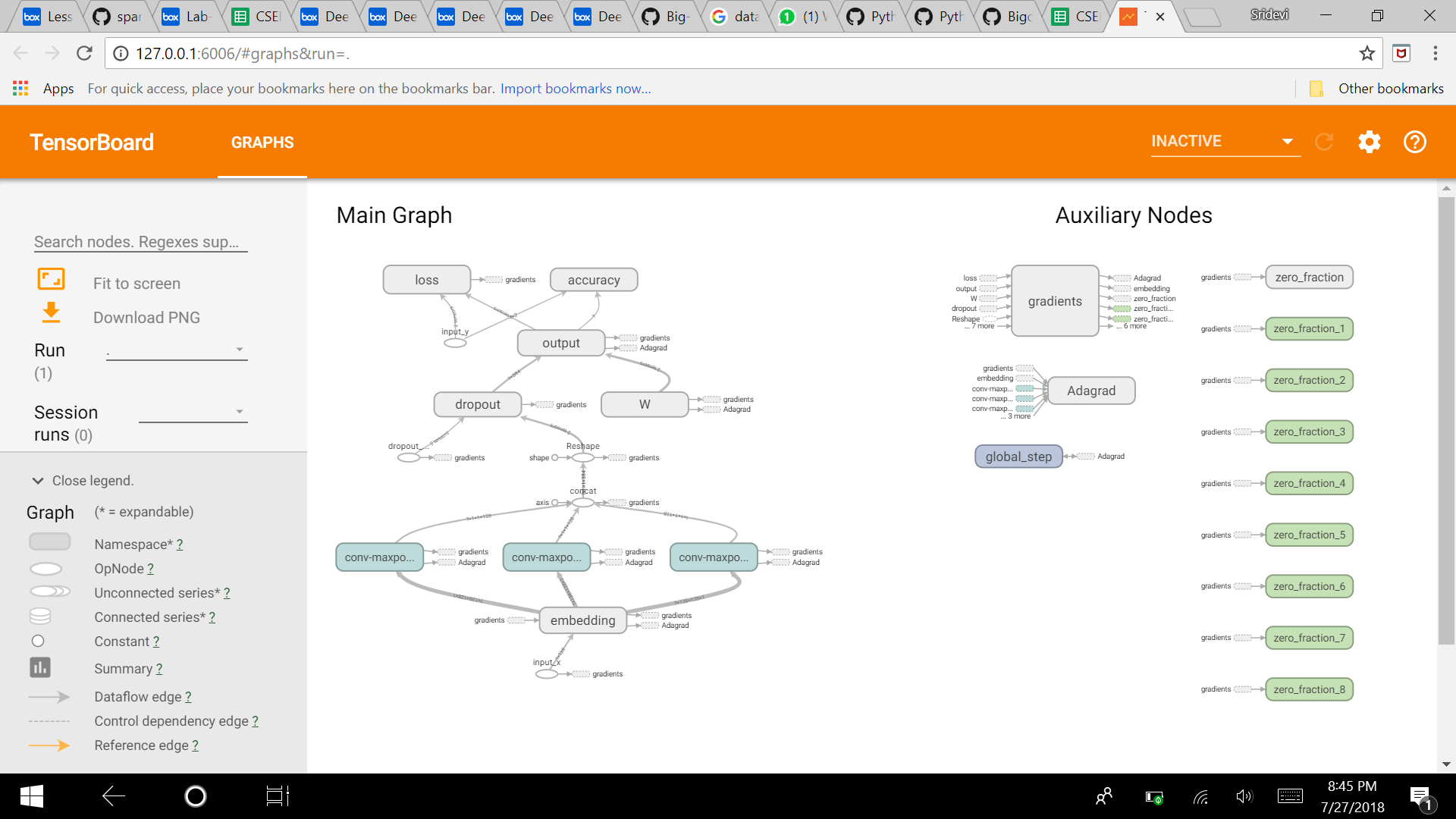
Story1.txt

Story2.txt

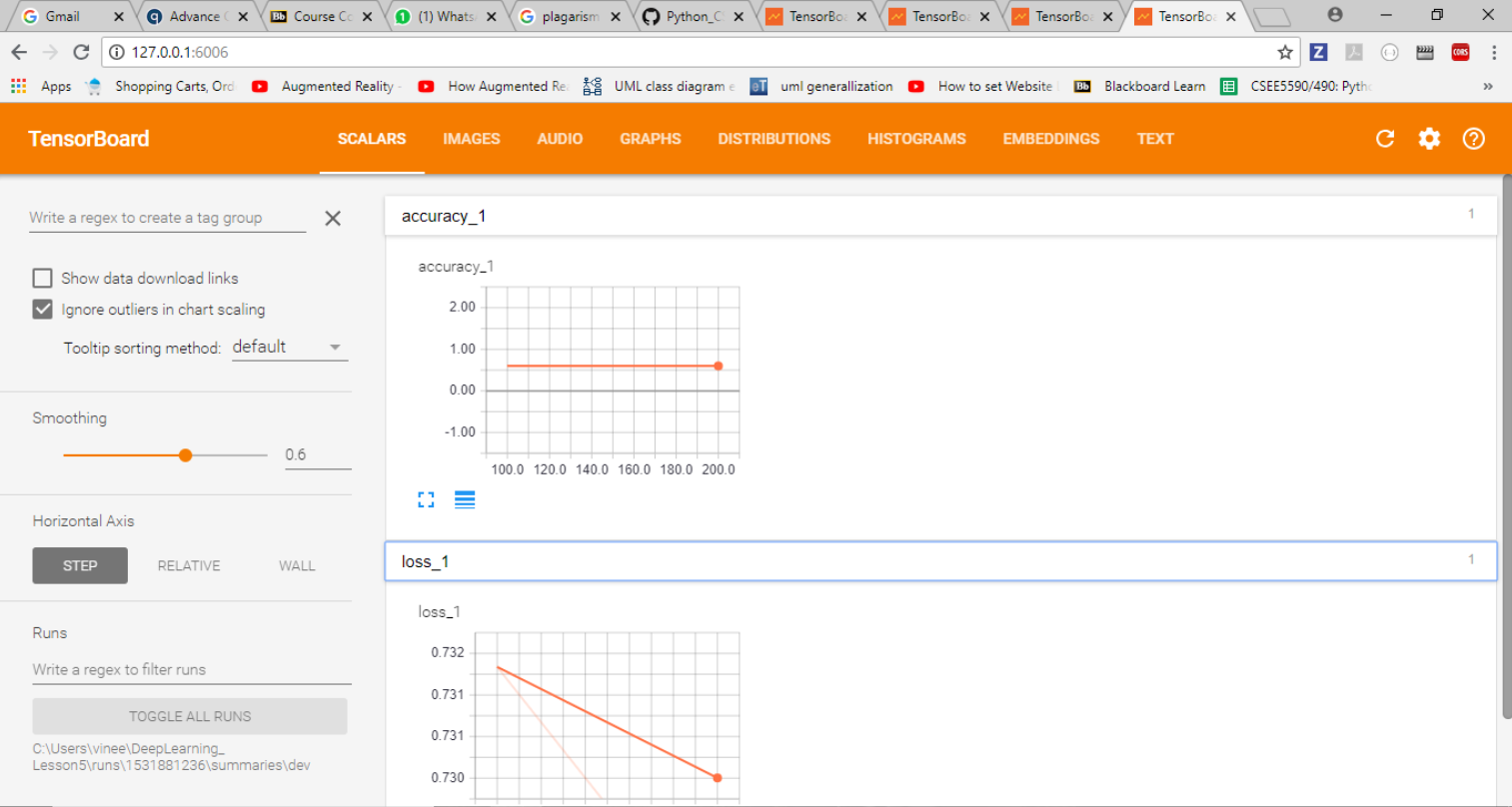
**Output Screenshot**:

****

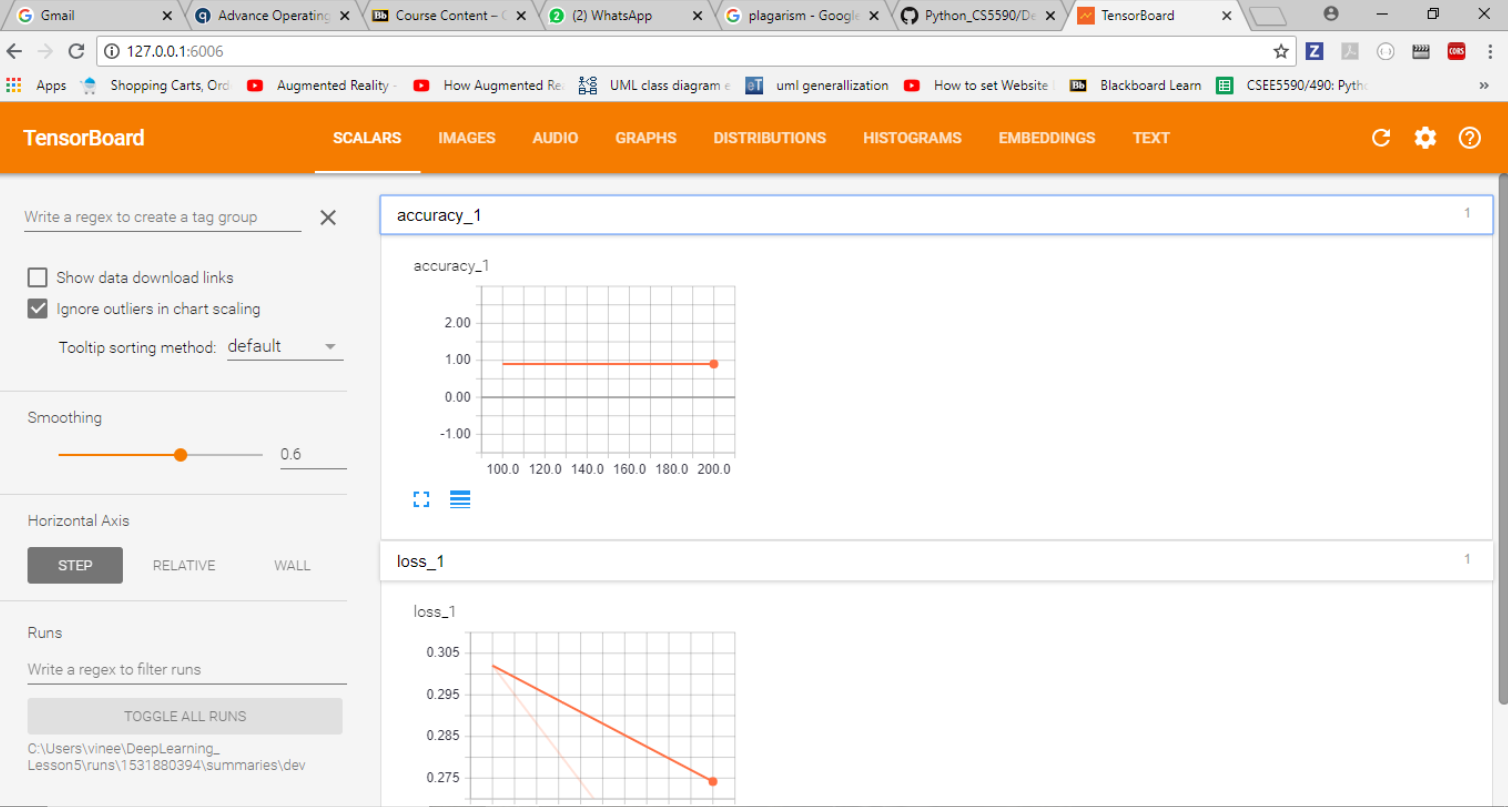
**Tensor board Graph:**

****

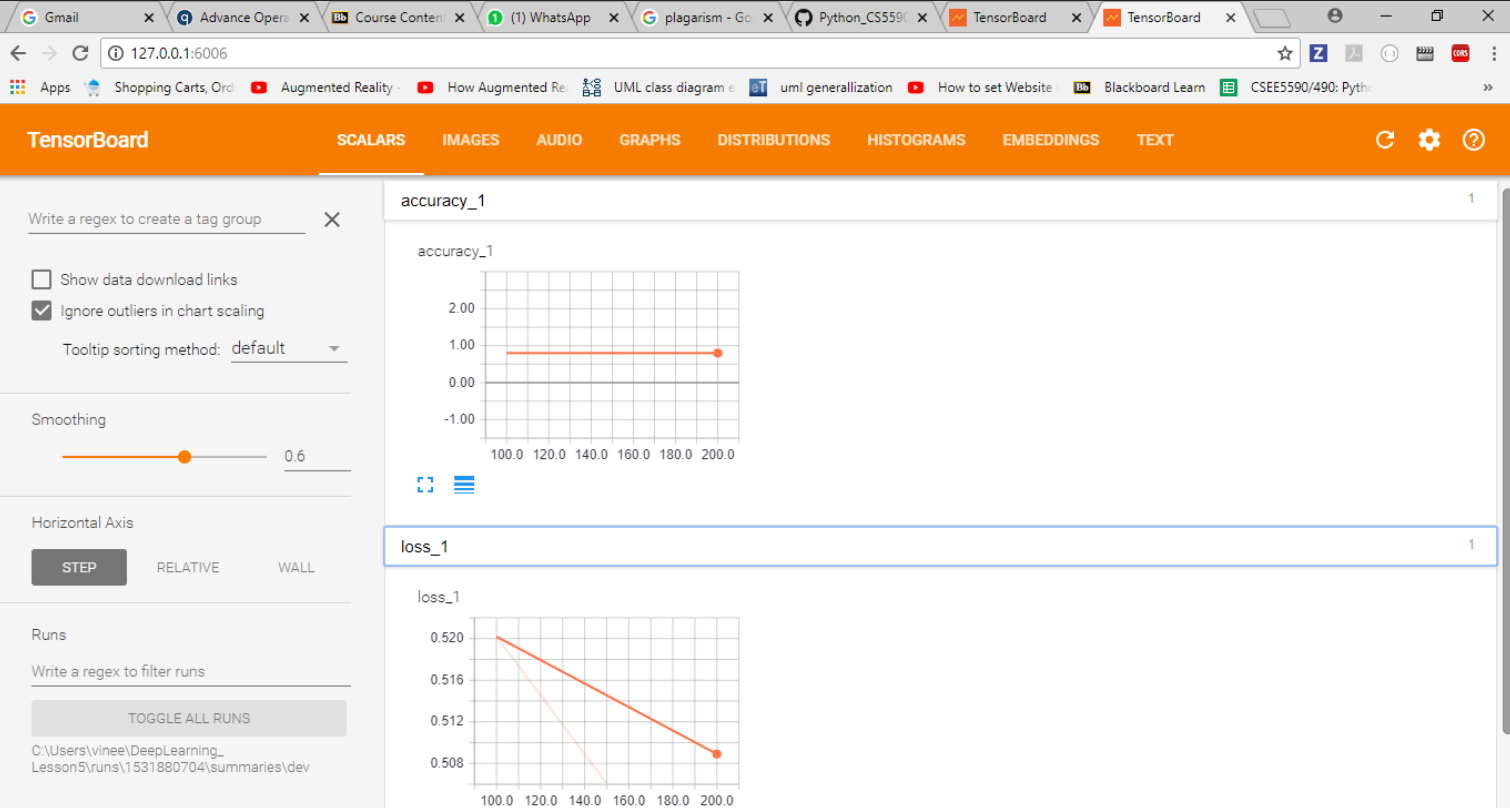
**RMS Pro Loss and Accuracy:**

****

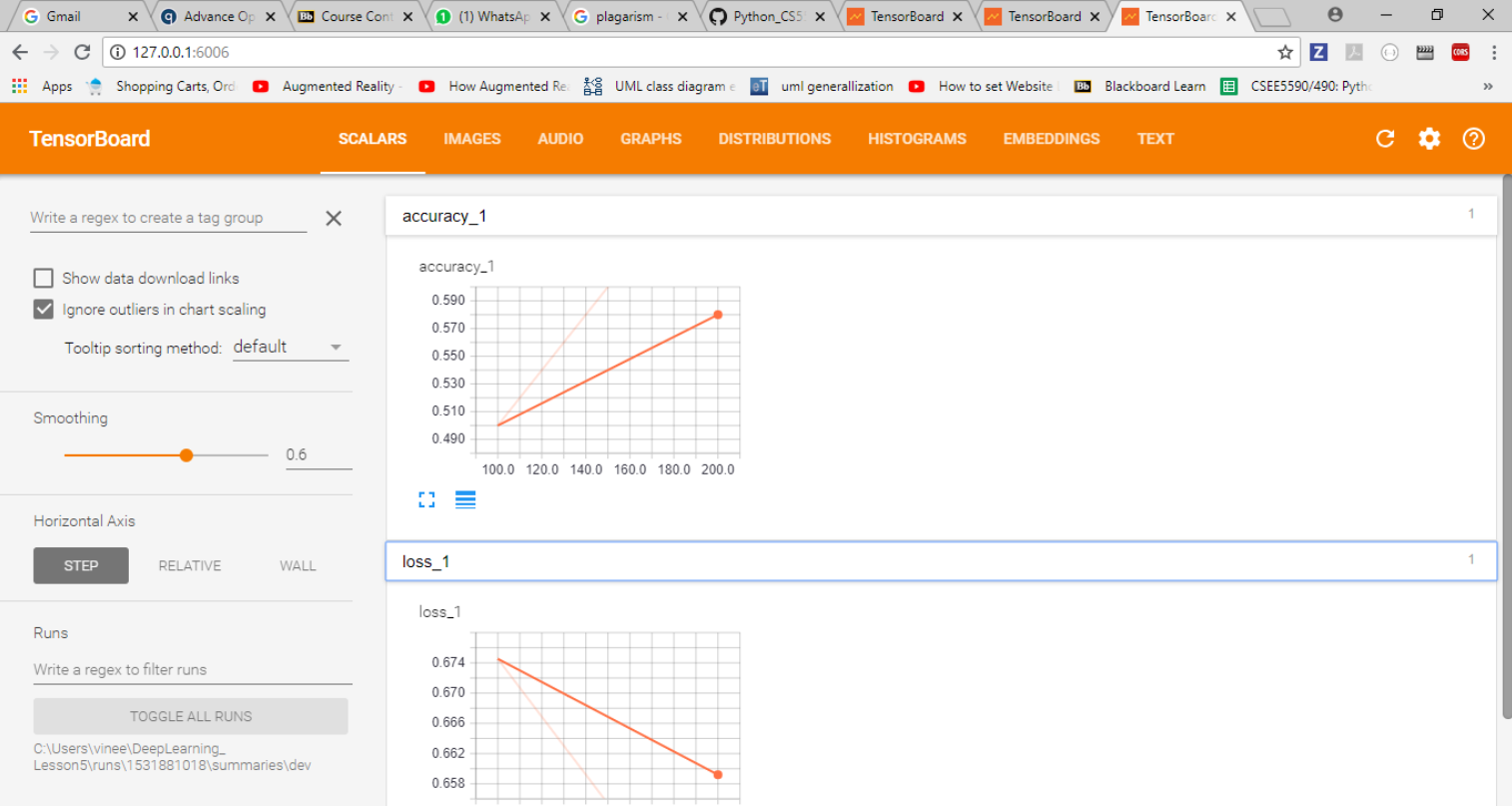
**Adam Loss and Accuracy:**

****

**Adagrade Loss and Accuracy:**

****

**Gradient Descent Loss and Accuracy:**

****

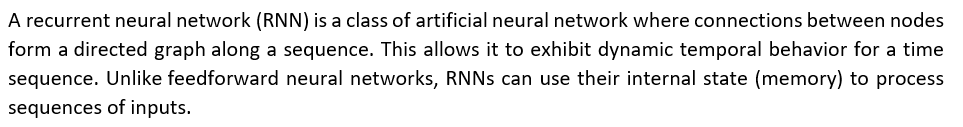
**Question 2:**

****

**Solution:**

This snippet code provides the above implementation which for the logistic regression, have considered iris data and used pandas to set into the data frame and installed sklearn package to load the iris data. After the data framing I have implemented logistic regression and calculated loss function to analyze the accuracy. In this data is divided into training and testing set such as 80% of training data and 20% of testing data. At last tensor flow session is created for the logistic regression model and calculated the accuracy

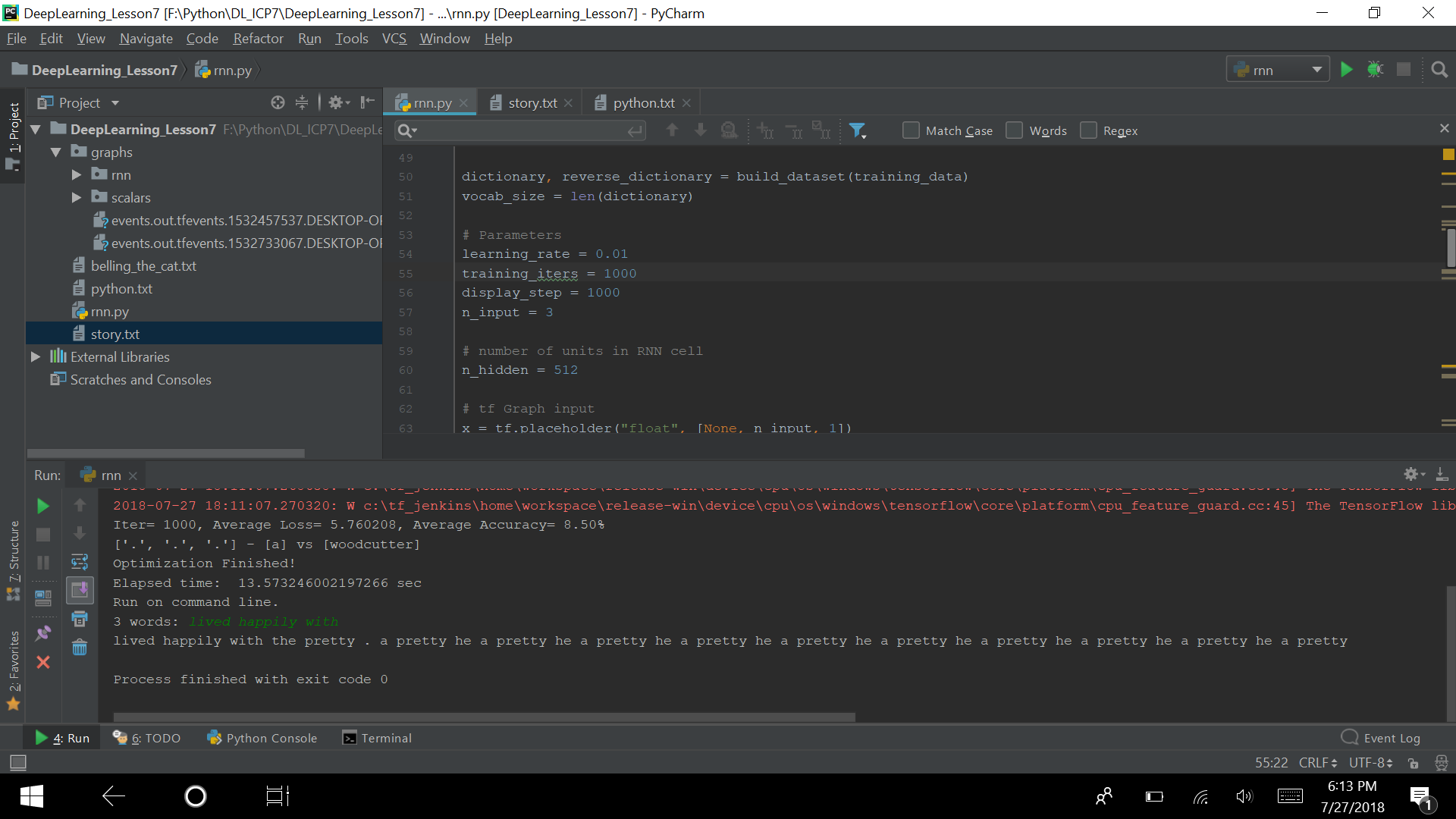
**RNN Text Classification:**

****

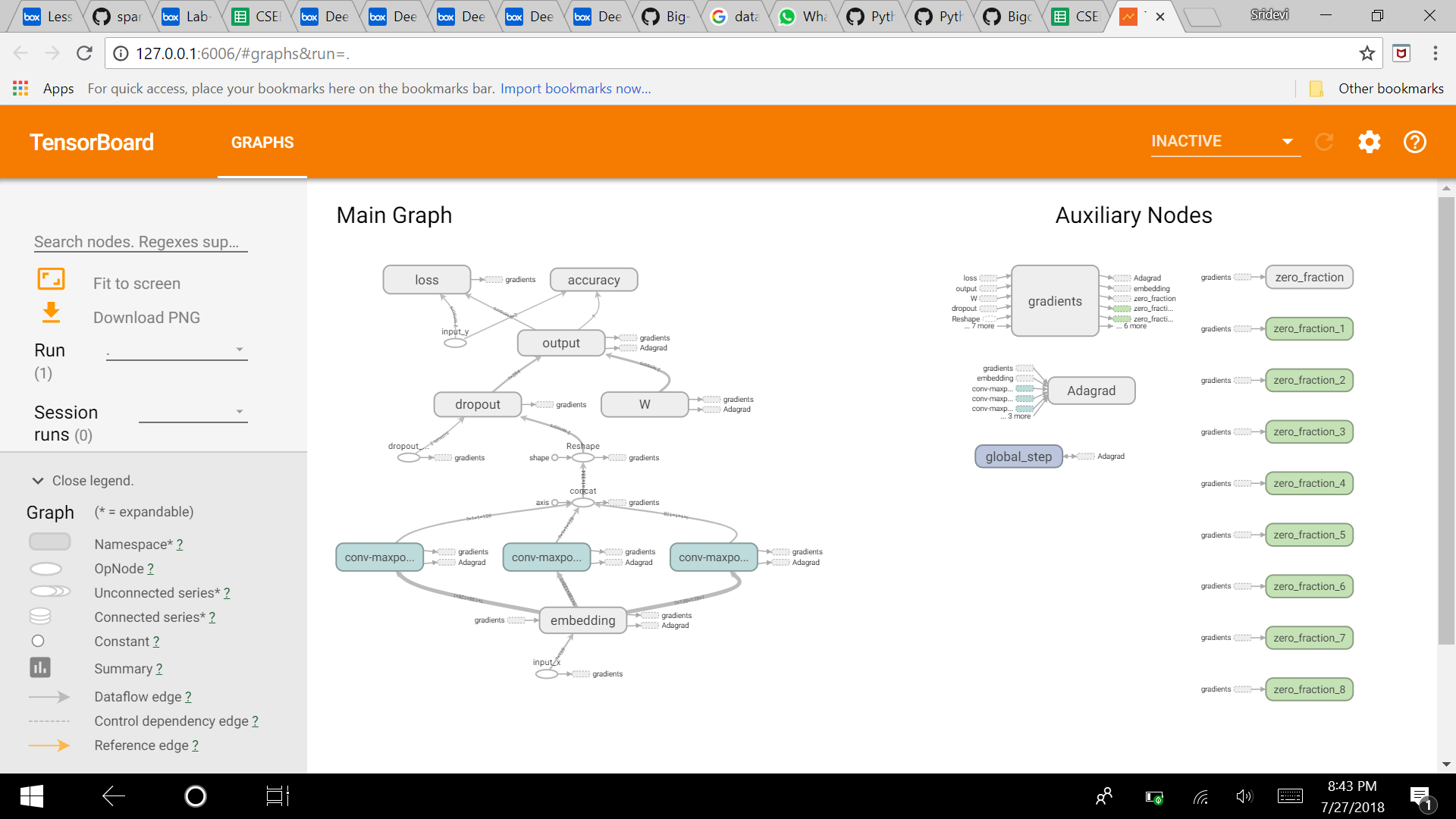
**Data Set:**

Story.txt

**Output Screenshot**:



Tensor board Graph:

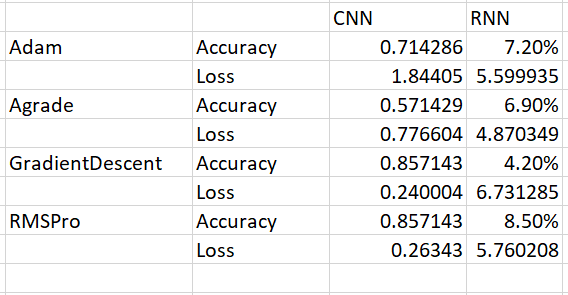


**Question 3:**

****

**Solution:**

Comparison table for CNN and RNN text classifications with different optimizers.

****

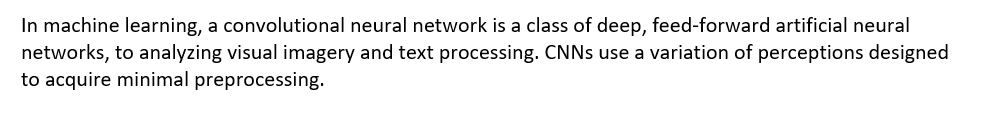
From the observations we could say that for both CNN and RNN RMS Pro Optimizer is giving better accuracy and less loss. By comparing the results from above table with respect to Loss, we could say that CNN outperformed compared to RNN.

**Question 4:**

**A close up of a logo

Description generated with high confidence**

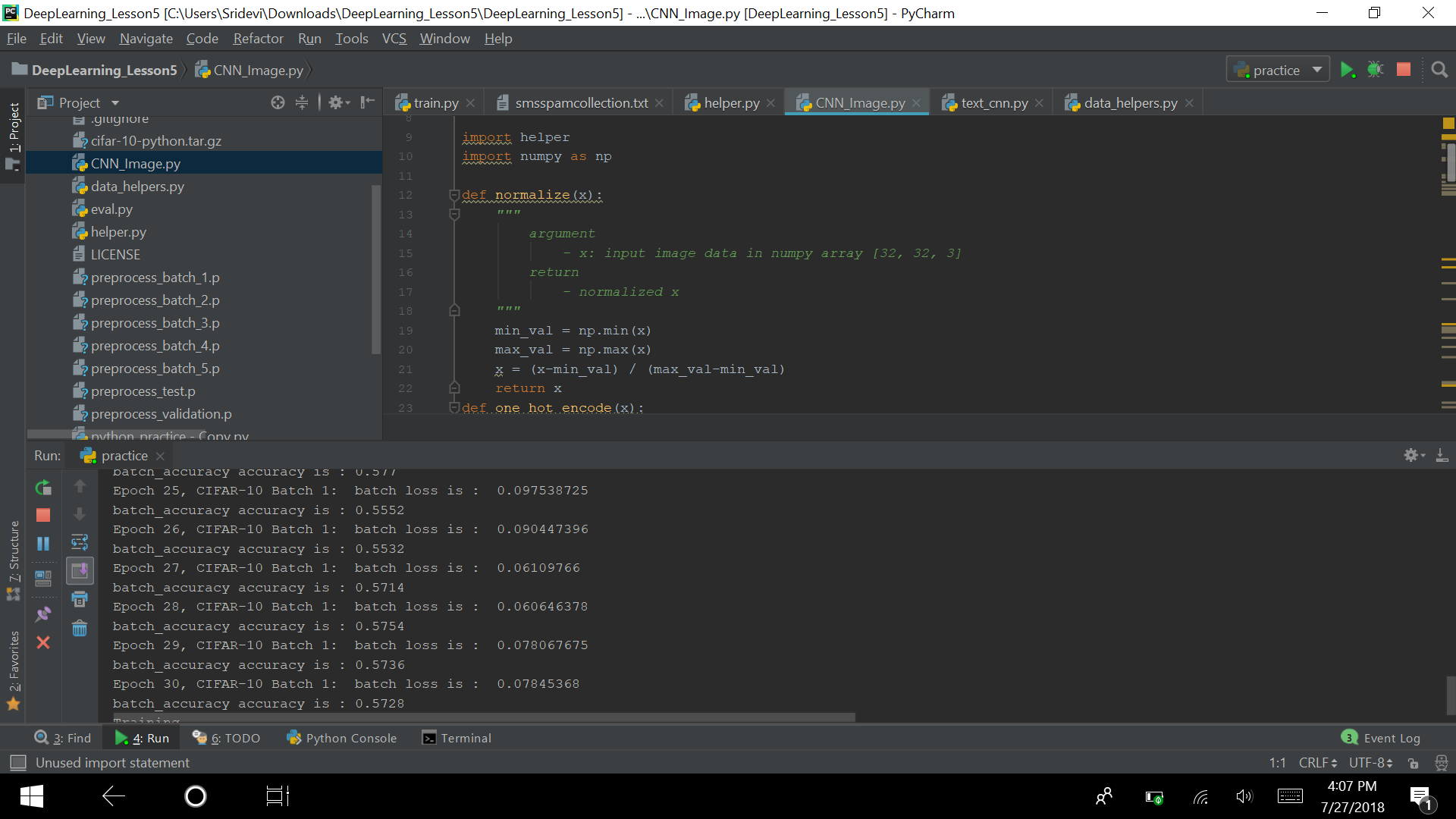
**CNN Classification:**

****

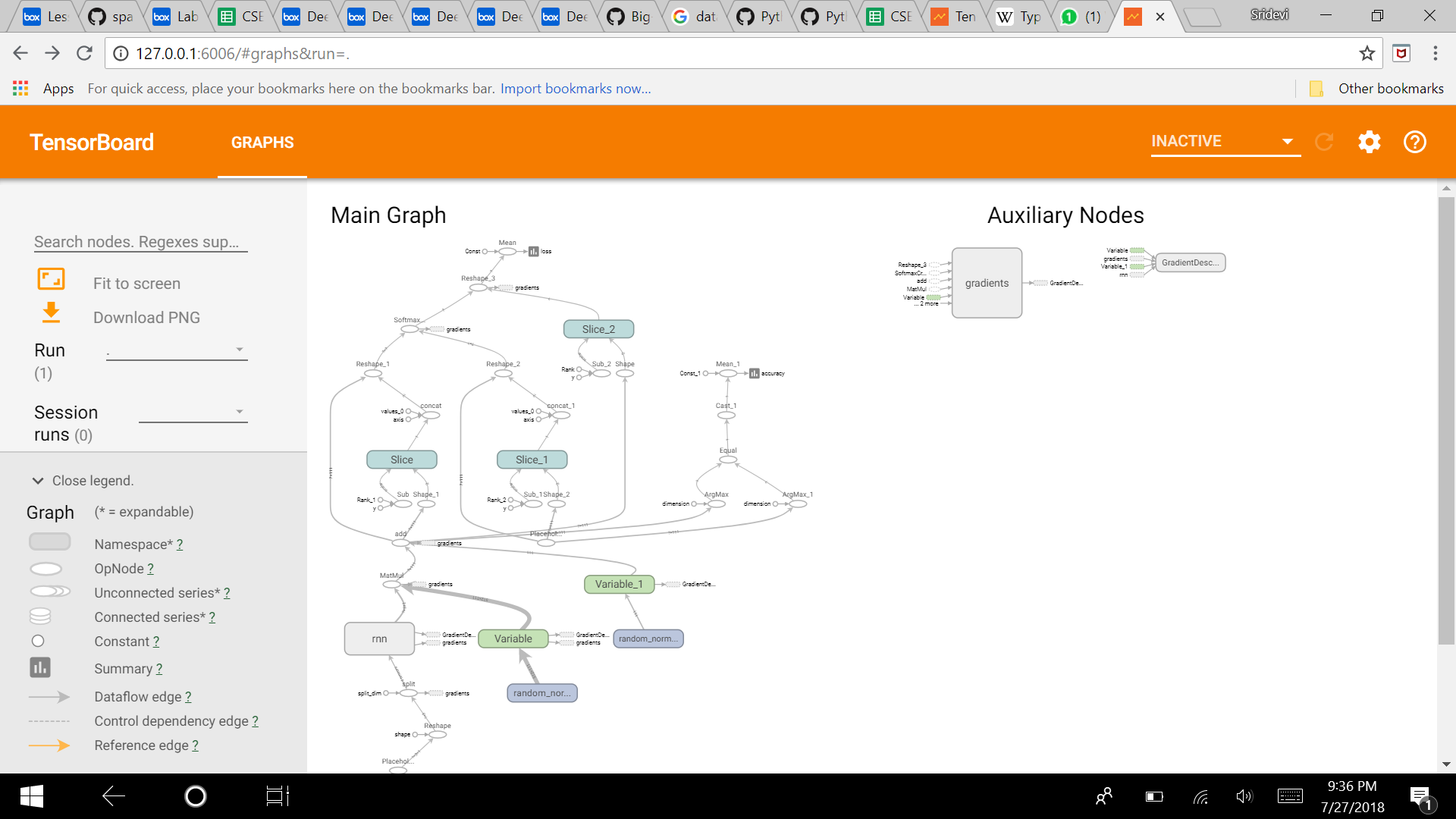
**Data set:**

Cifar-10

**Output Screenshot**:



**Tensor board Graph:**

****

**Source code**

**Video Link:**

**Wiki Link:**