PYTHON LAB ASSIGNMENT 2

BY

Team 2

Akhil Teja Kanugolu – 16297766

Geetanjali Makineni – 16290659

Github Link: <https://github.com/akhilkanugolu/Lab2_Python>

Video Link: https://umkc.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=dd2d29bf-e03b-43a6-b955-abb80047ab7f

Wiki Link: <https://github.com/akhilkanugolu/Lab2_Python/wiki>

**Introduction:**

In this lab we did programs that are related to some concepts like Word Embeddings and different Loss functions as well as Optimizers and also Convolution Neural Networks, Autoencoder, LSTM, Implementing Text Generator and Dimension reduction on the image. Finally, we did TensorBoard and TensorFlow graphs and sessions.

**Objectives**

The main objective is to implement some programs that will cover all the concepts that are mentioned above. If required we also compare and also contrast the two methods providing the respective result based on the absolute solutions we got obtained.

1. Building one linear Regression Model here using Sequential model and showing the graph on Tensorboard and to change some parameters and then plot the loss.
2. Implementing the logistic regression and normalizing the data and then plotting the loss on the Tensorboard and then changing any of the three hyperparameters as well as reporting the accuracy.
3. Image Classification by usage of CNN.
4. Text Classification by usage of CNN.
5. Text Classification by usage LSTM.
6. Comparing and reporting which one of the model is the best for Text Classification here by tuning the respective hyper parameters.
7. Encoding and also decoding by using the Autoencoders on one particular image.

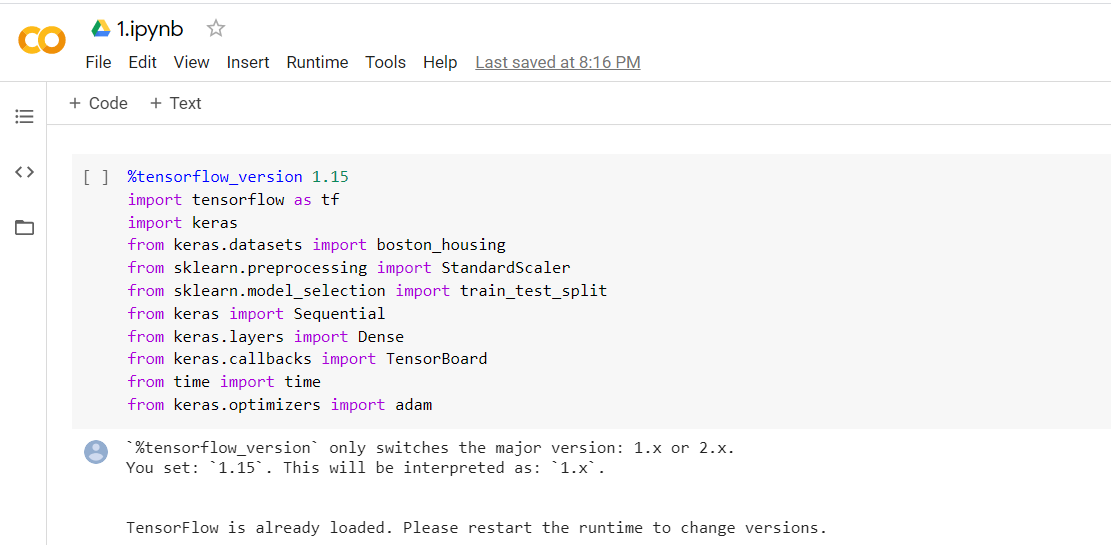
* **Datasets:**
* Boston\_housing
* Heart.csv
* Natural\_images
* Movie Reviews
* Mnist

**Workflow:**

We have used the Boston dataset here

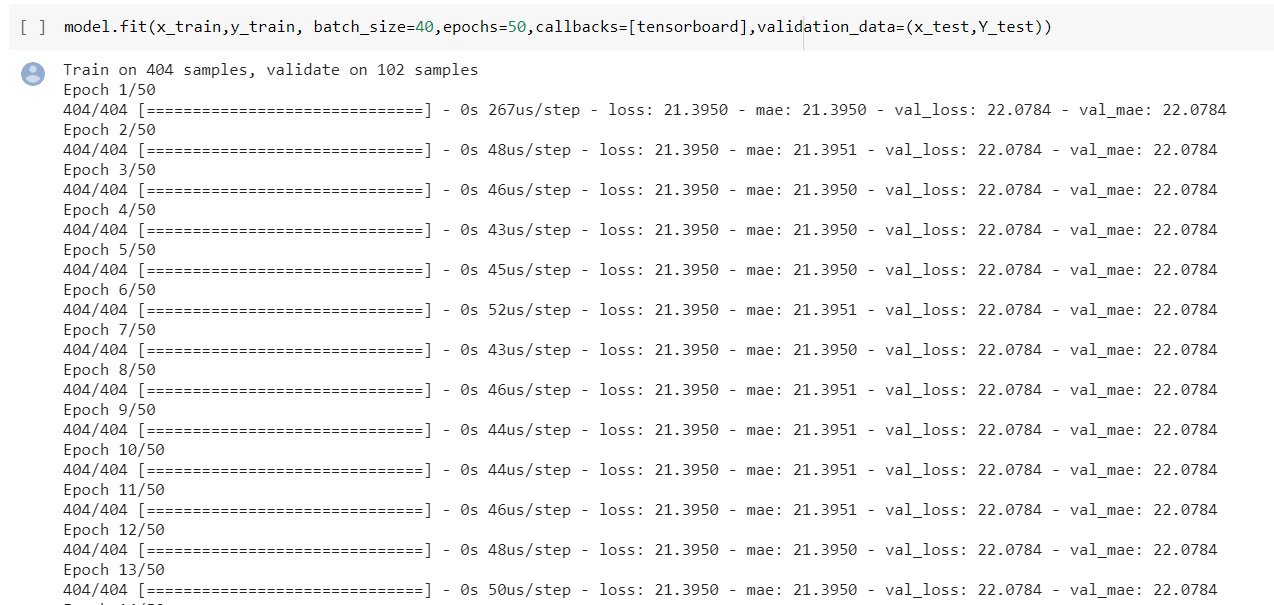
We imported all the packages required for processing the dataset.

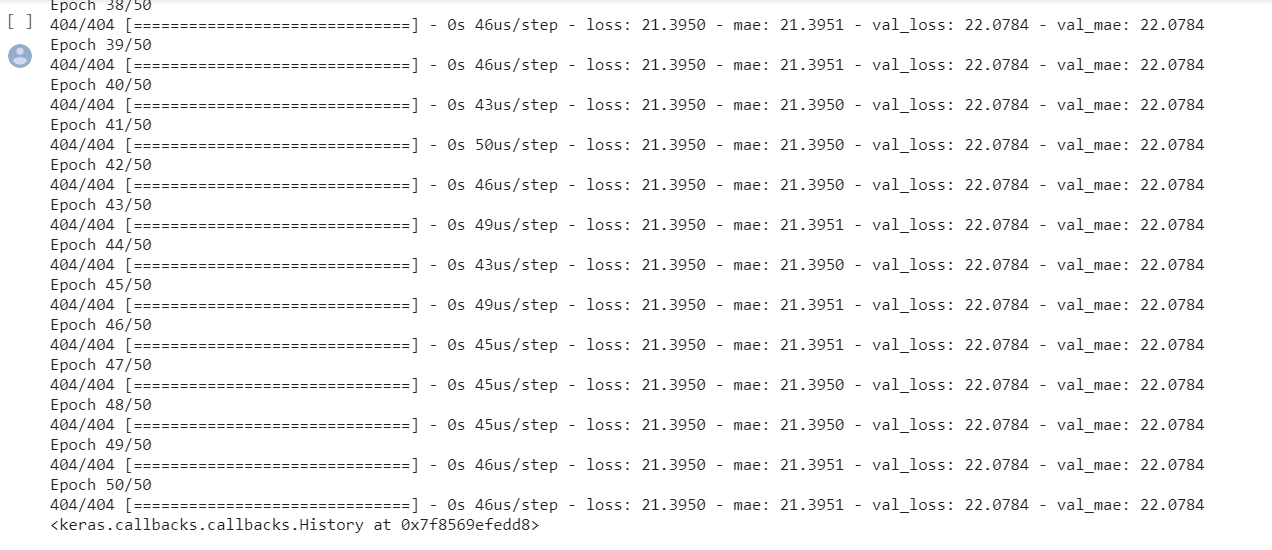
Task 1:

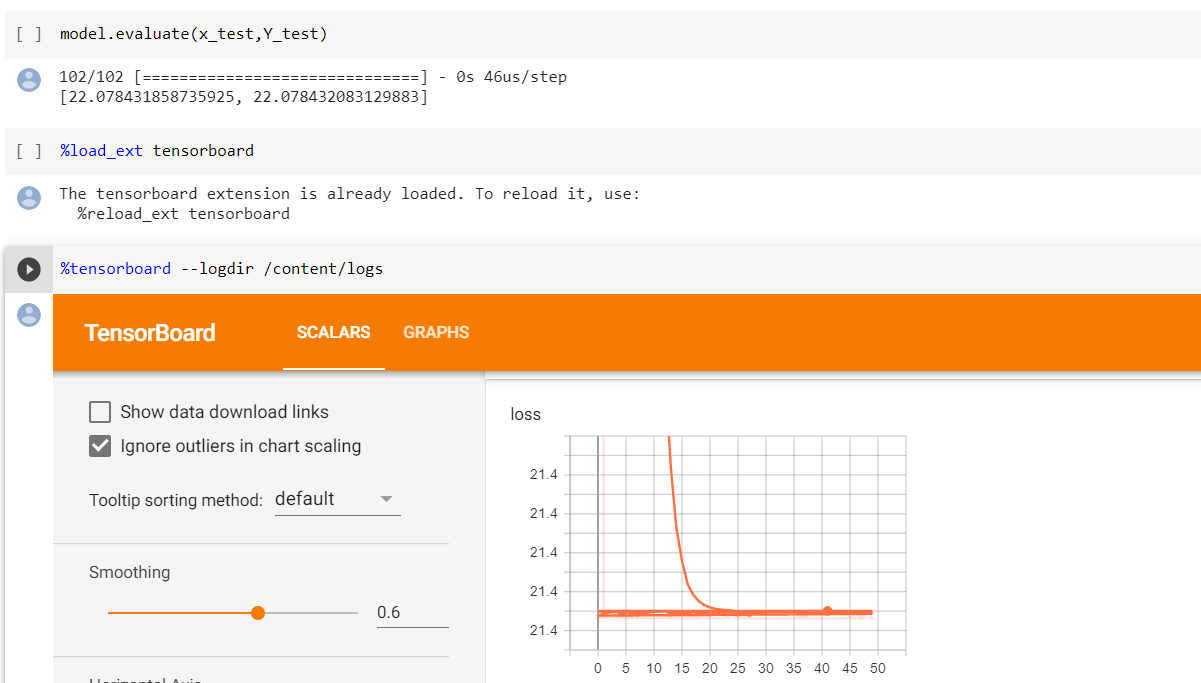


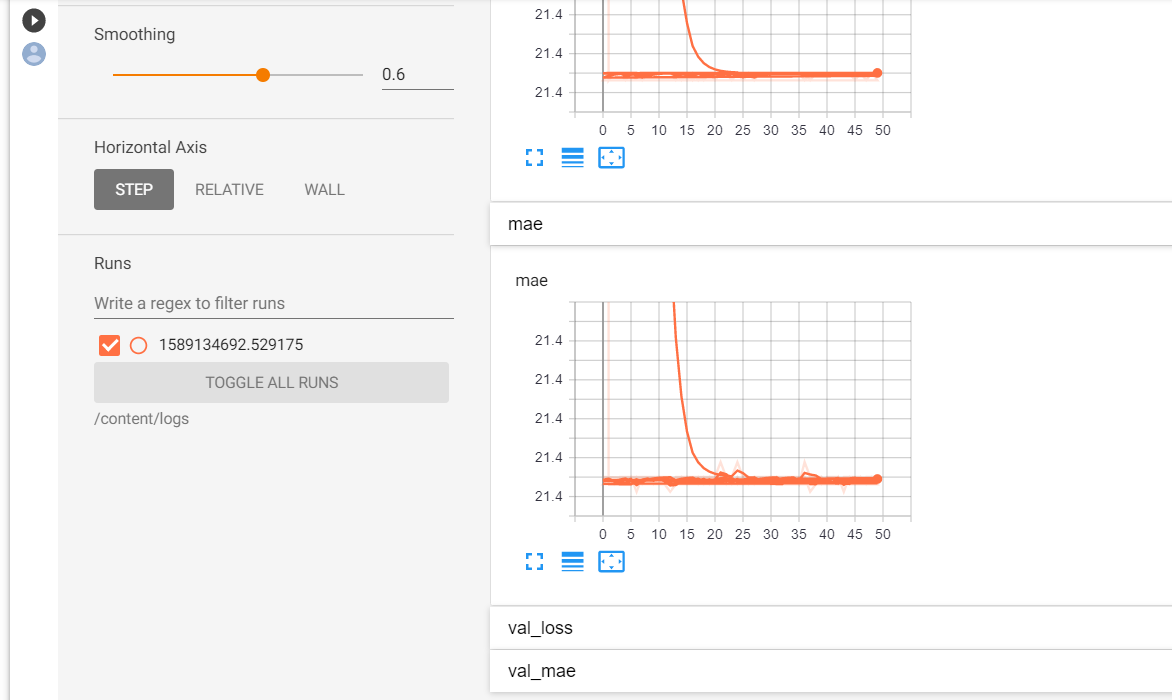


Fitted the model using epochs=50 and batch\_size=40 and used the optimizer =’rmsprop’ with sequential model



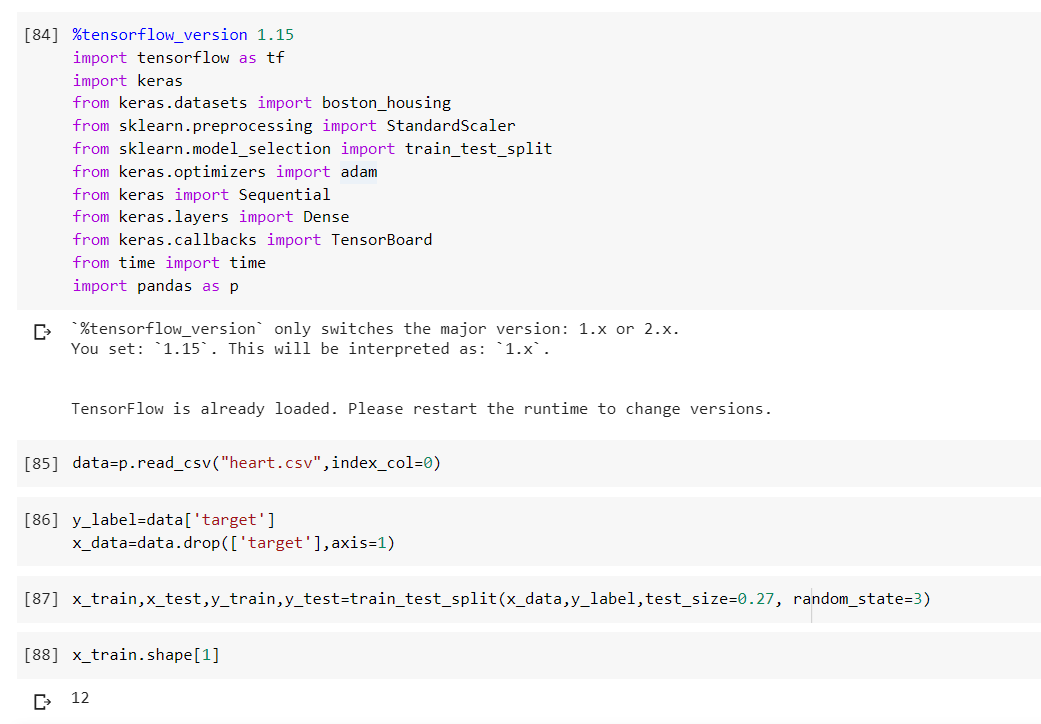
Finally visualized the loss and accuracy plots in the tensor board





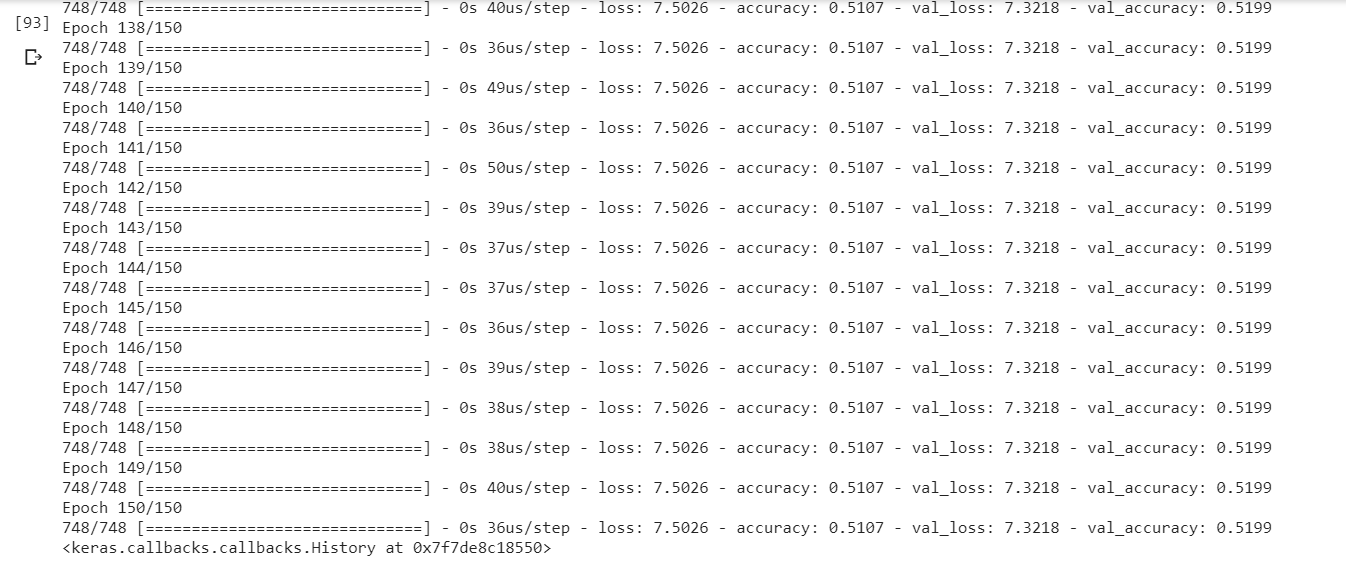
Task 2:

Heart.csv file was Imported later, other than target column was loaded in x & target column was loaded in y. later splitted with test datab size of 27%.

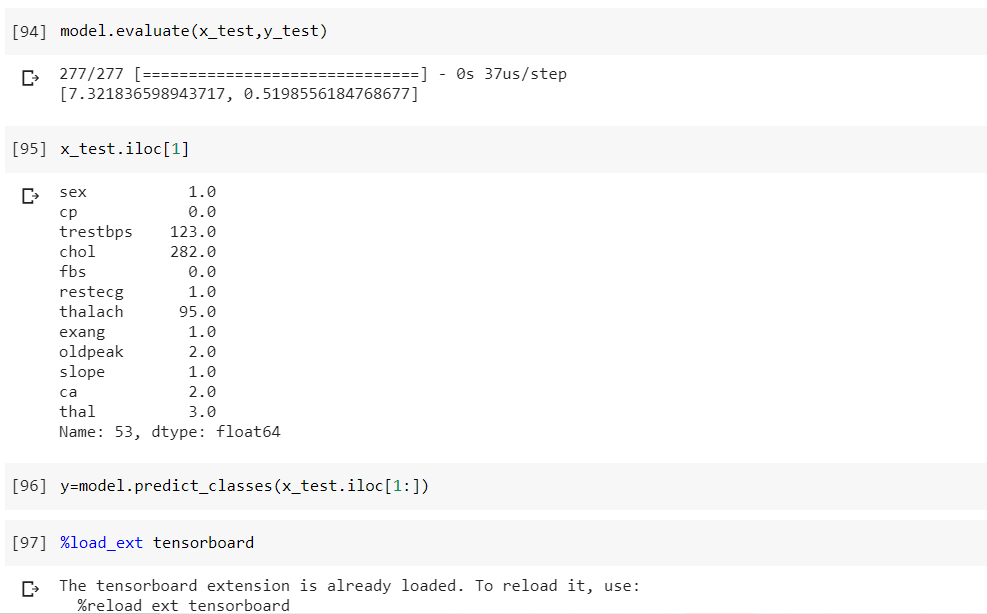


Later fitted to model with batch size=50, epochs=150

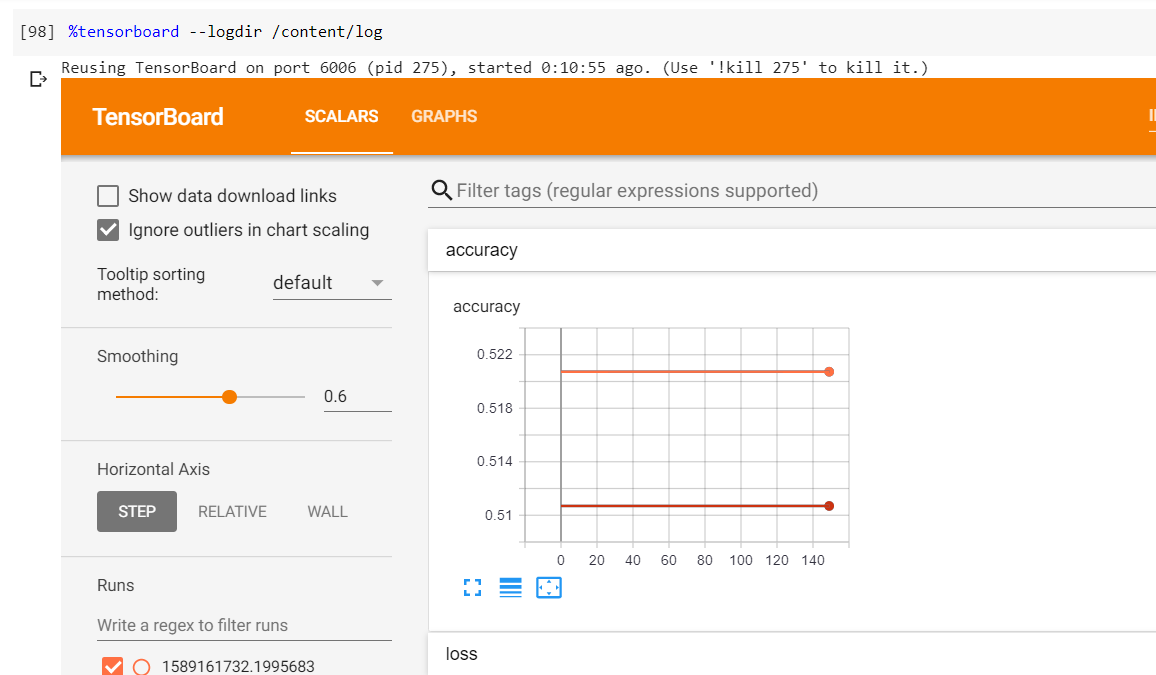


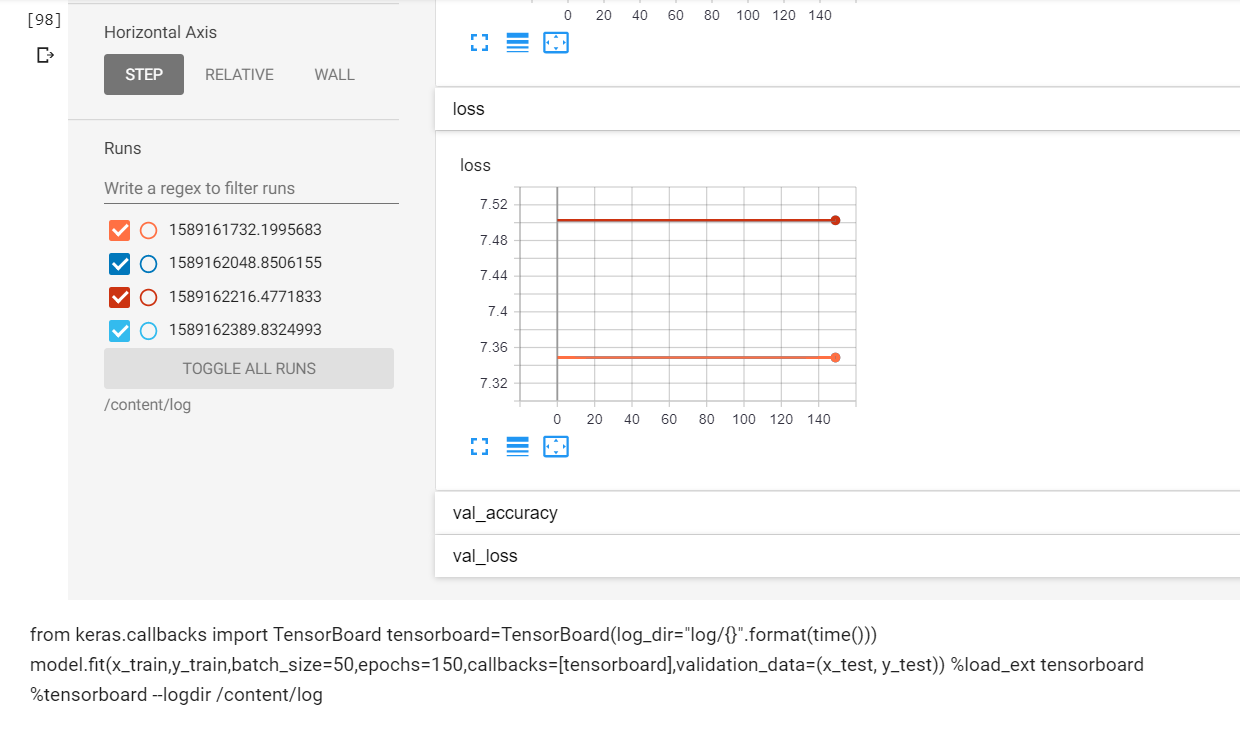


Evaluated the model where I got accuracy with 51%



Visualized the graphs using tensor flow. Here accuracy ranging more than 52%. And loss with 74%.

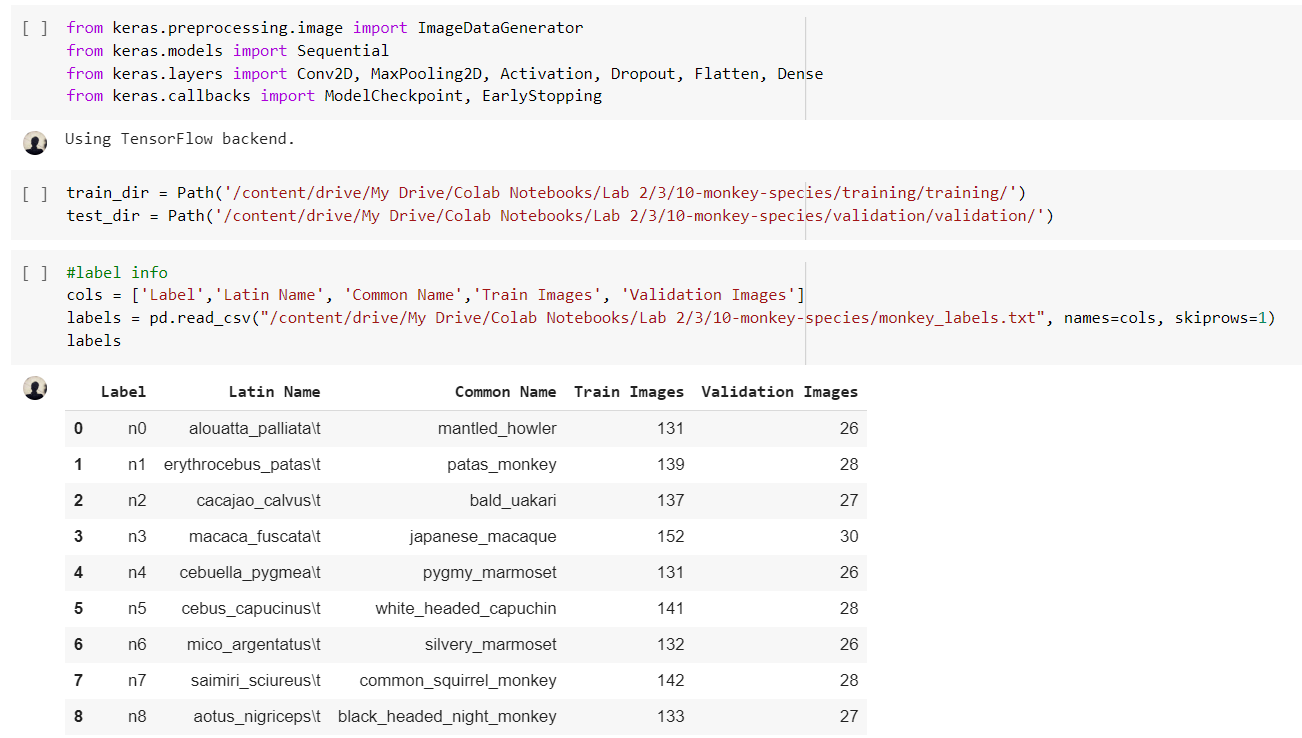


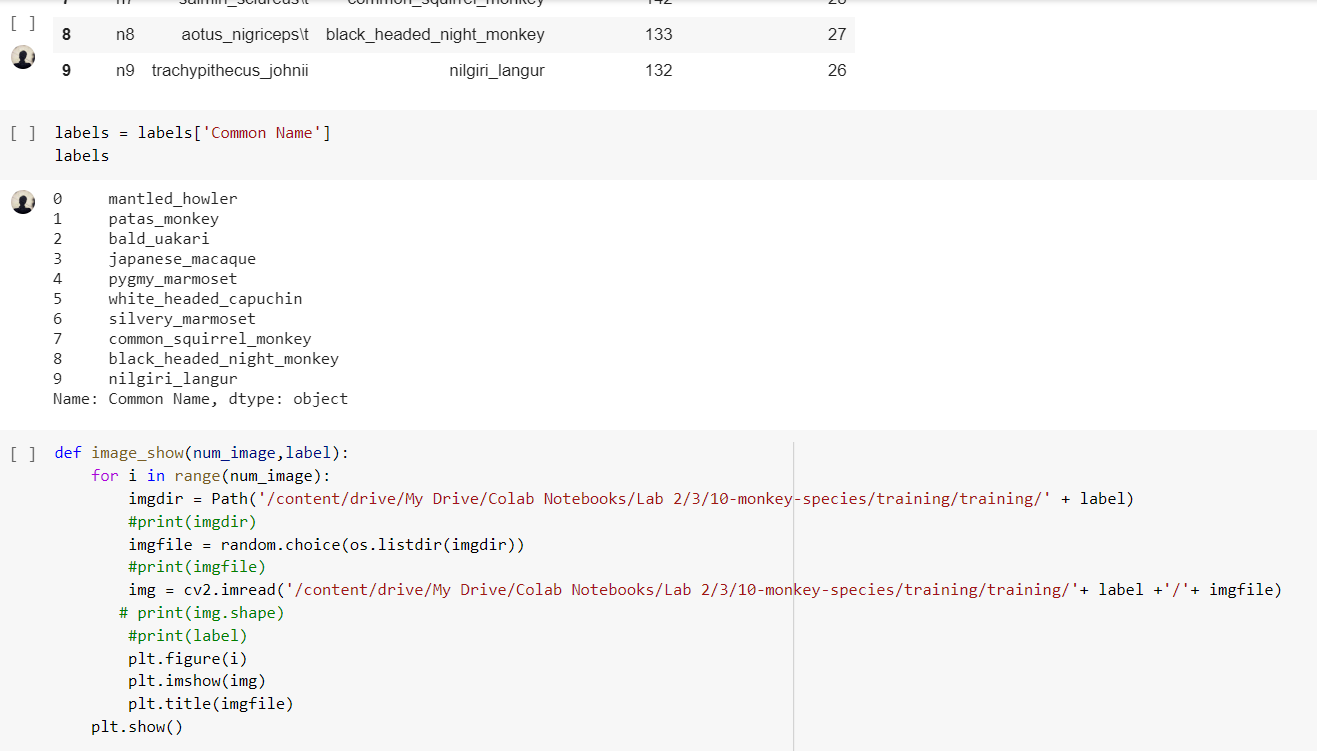


Task 3:

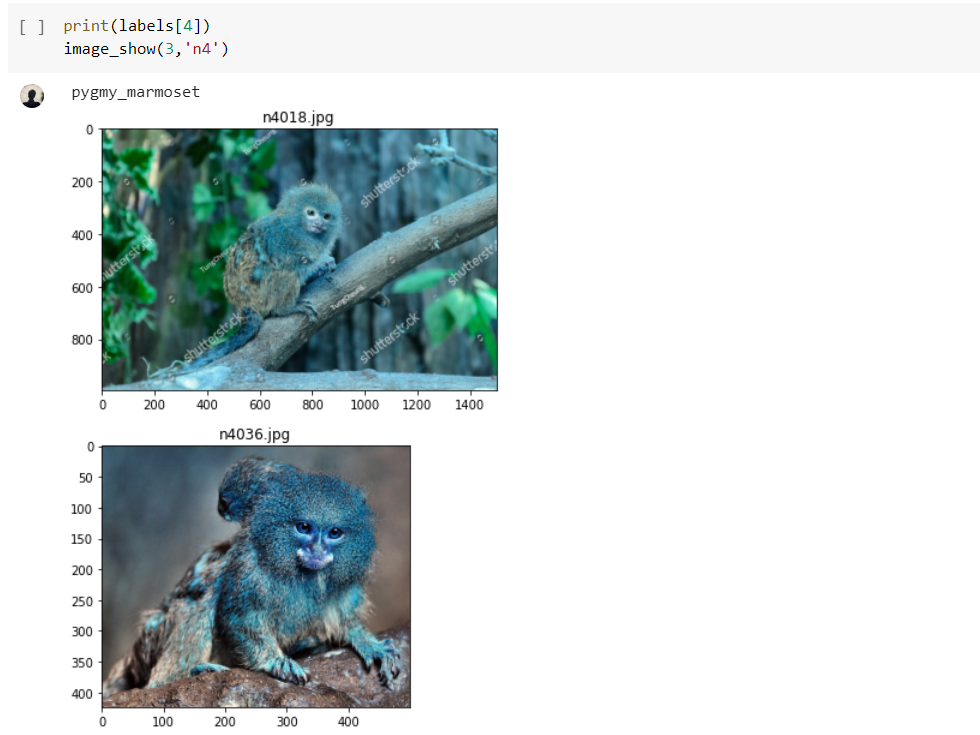
Mounted the drive. Later we imported the train and test data from respective paths. After this we pre processed the data by dropping several columns which have no impact on predictions.





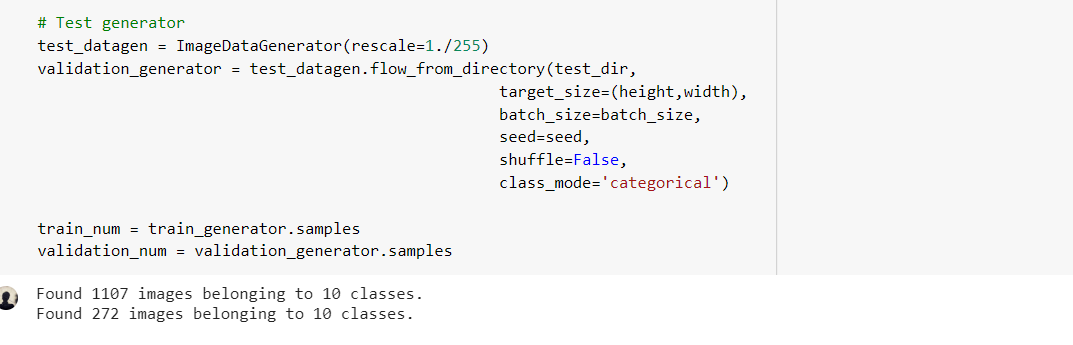


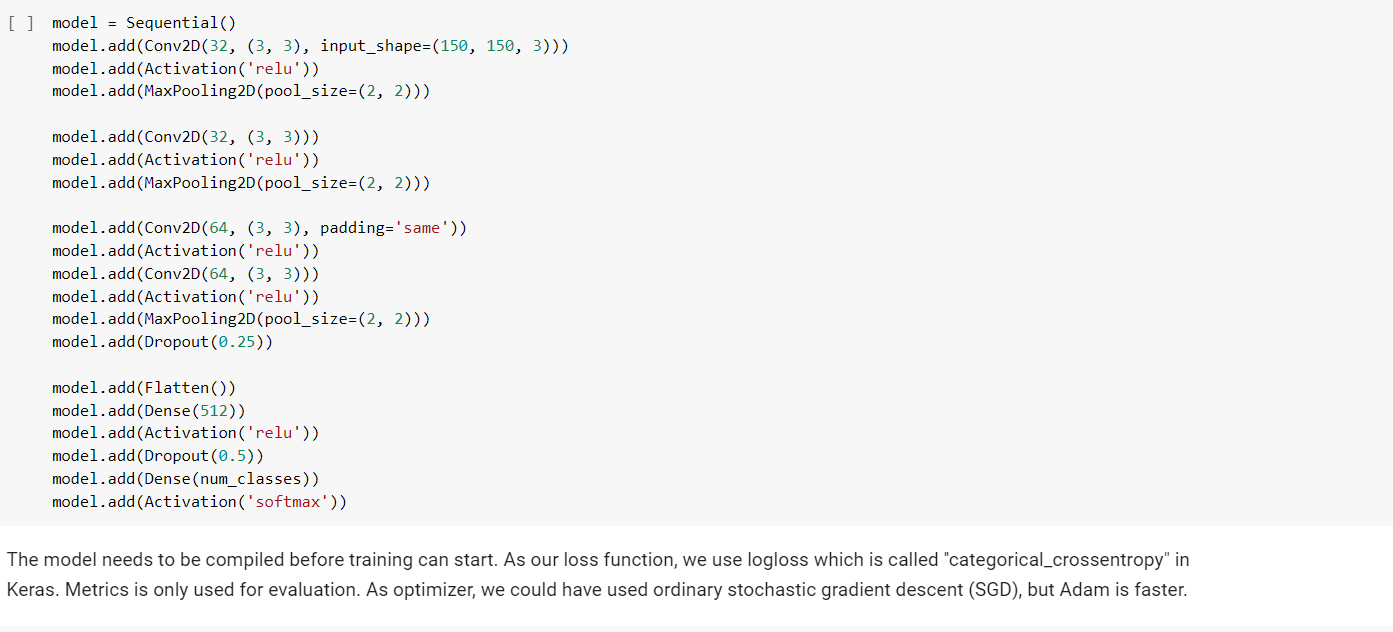
We have n1-9 types of monkeys so we checked the labels and checked some of the image of the test data.

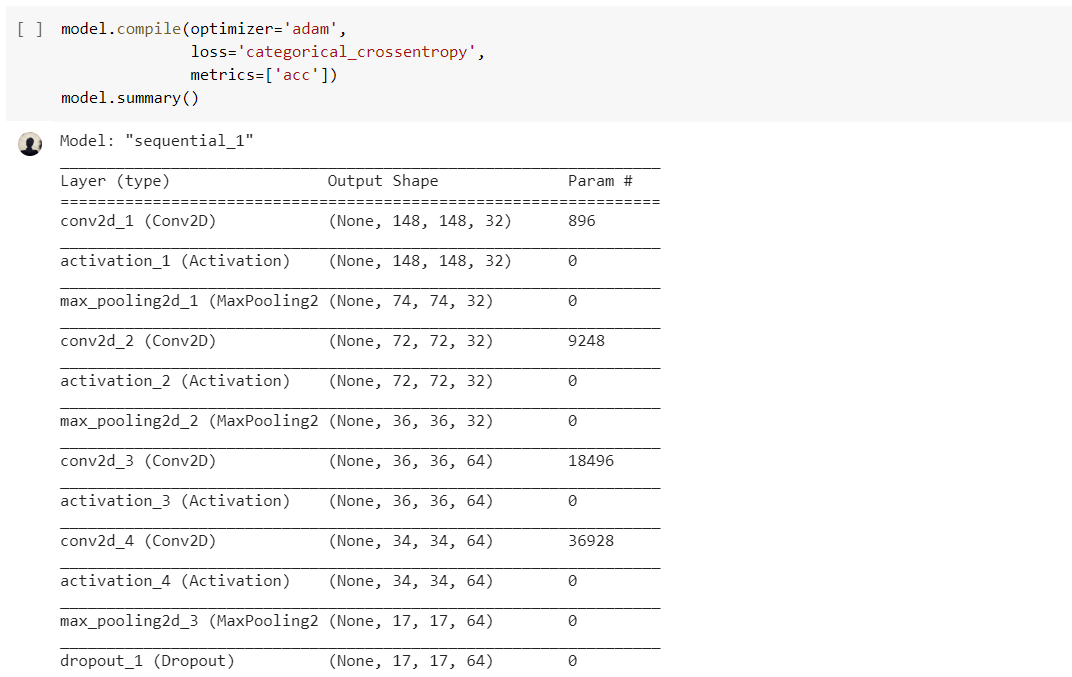


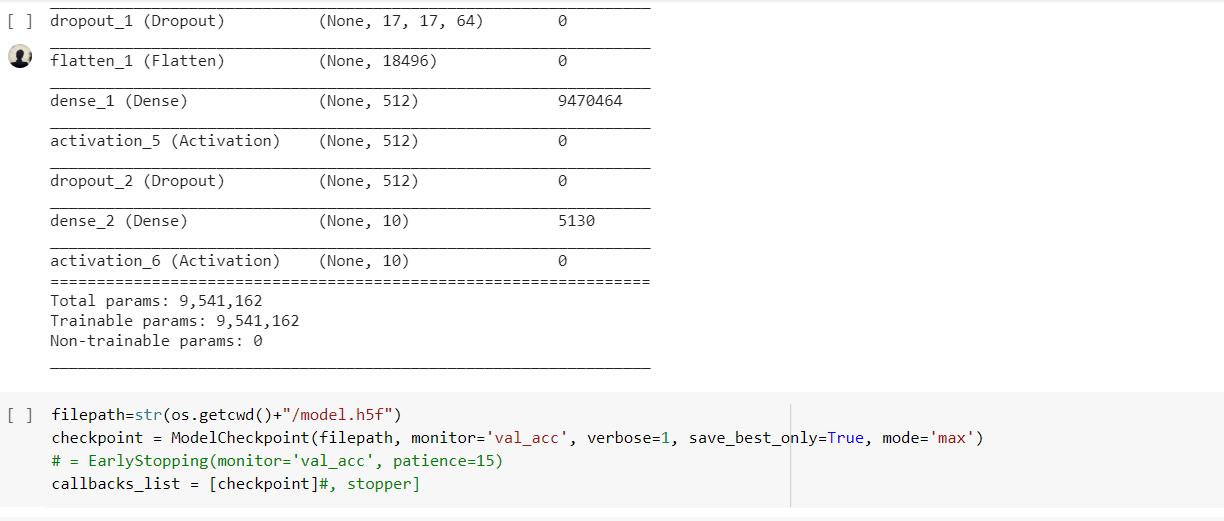


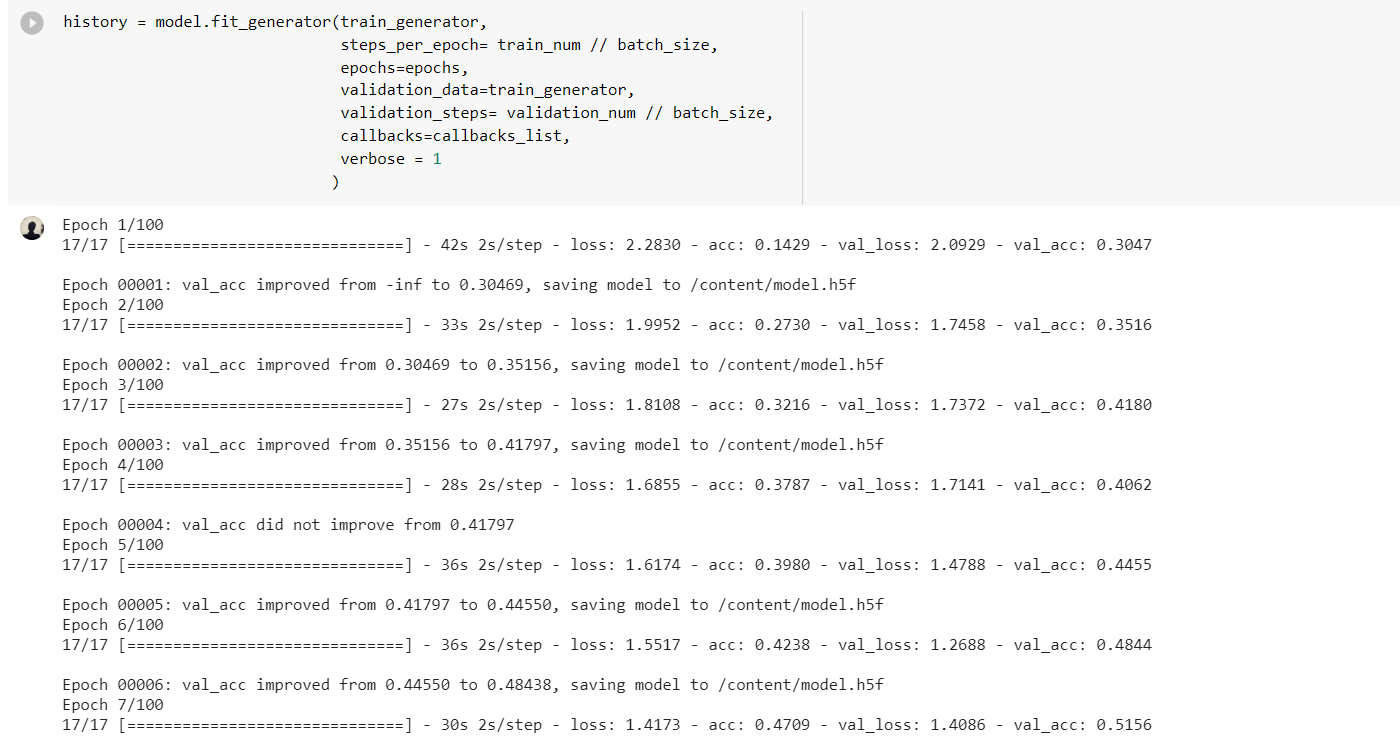


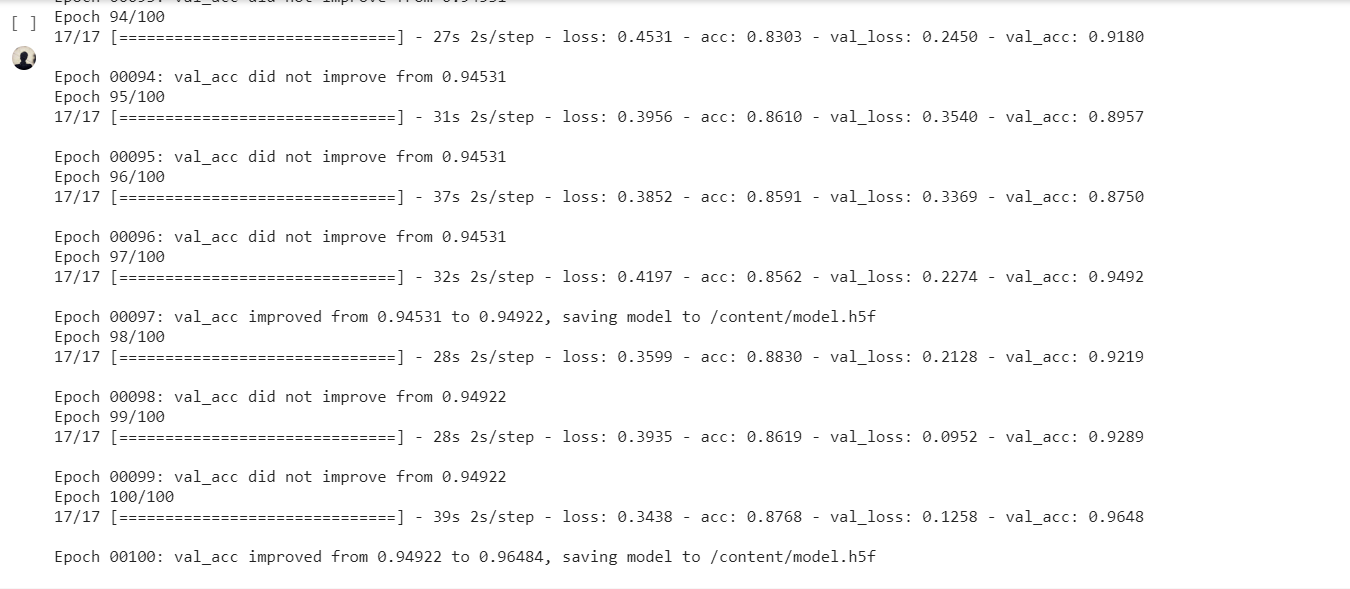




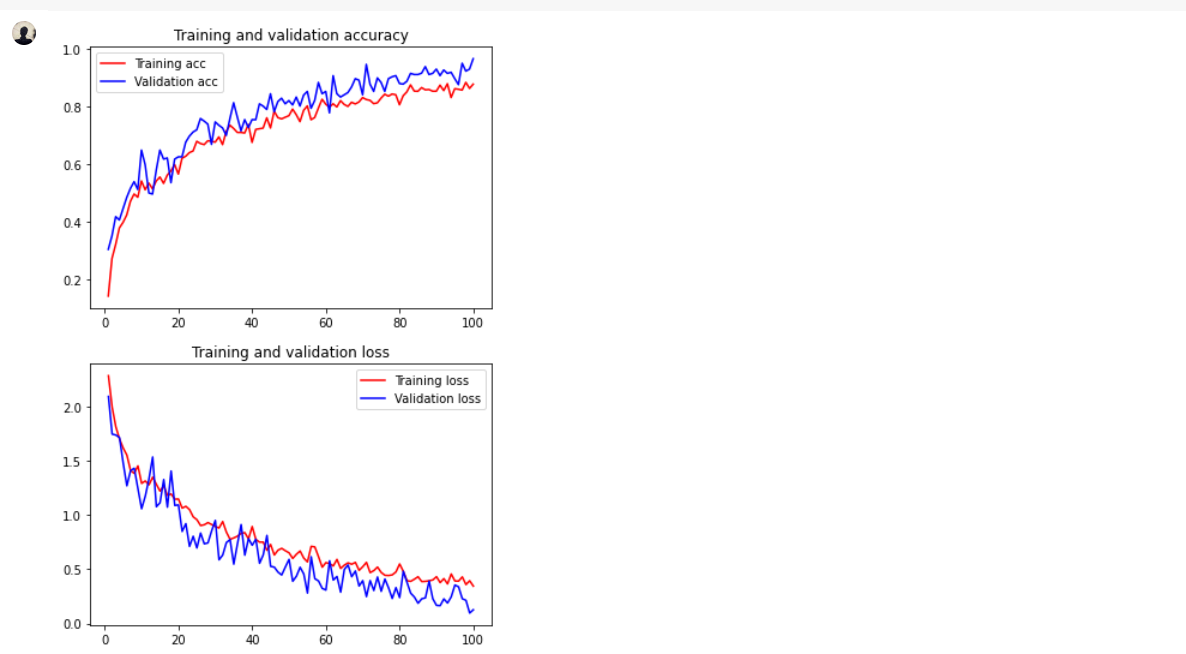


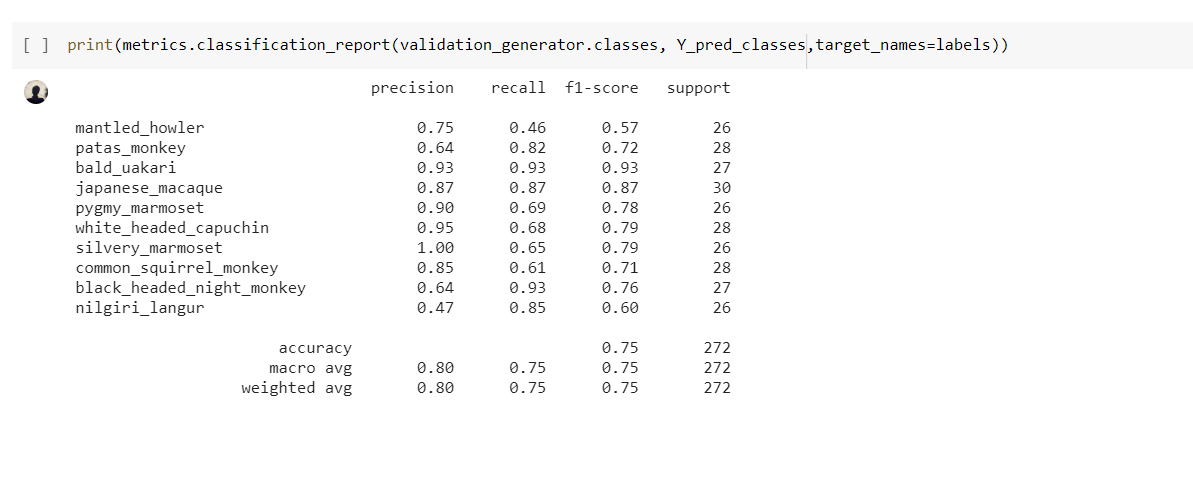






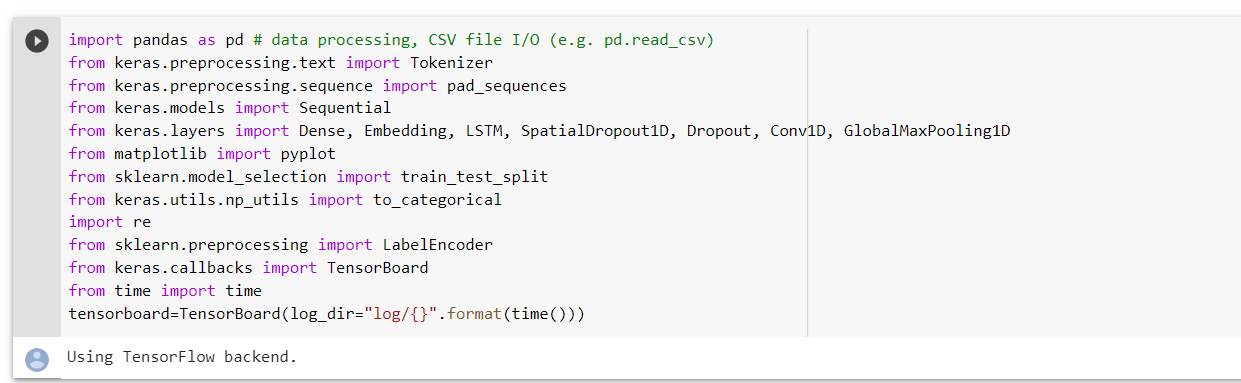






Task 4:

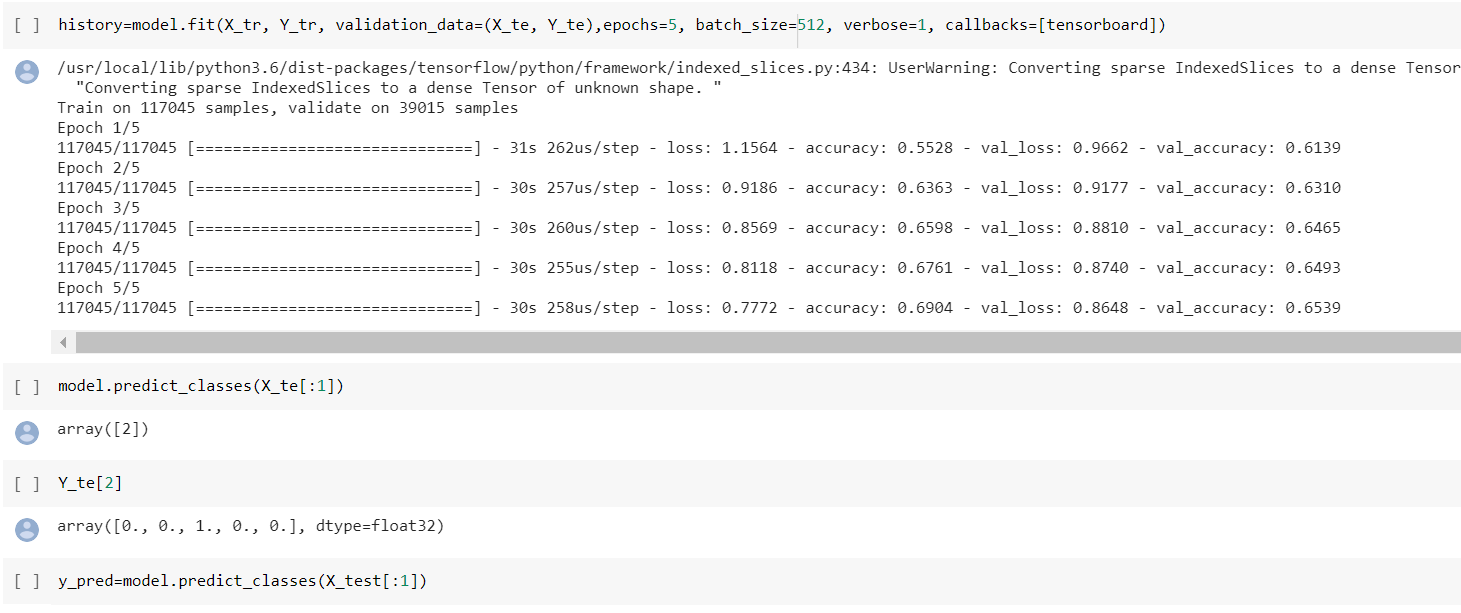
Text Classification using CNN



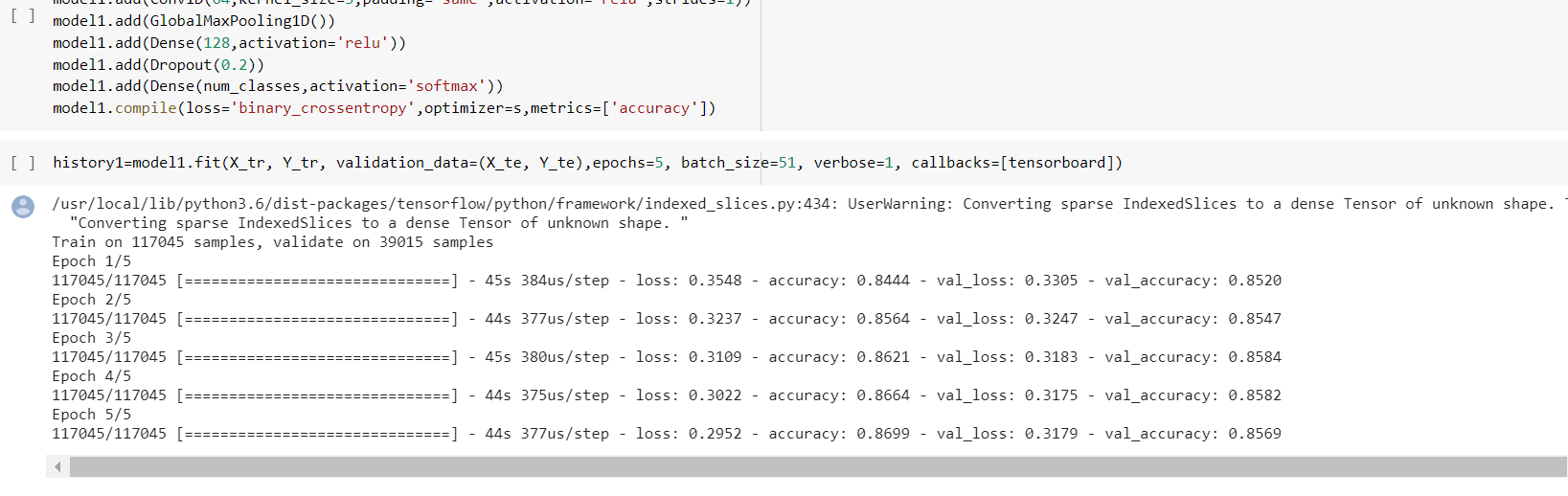








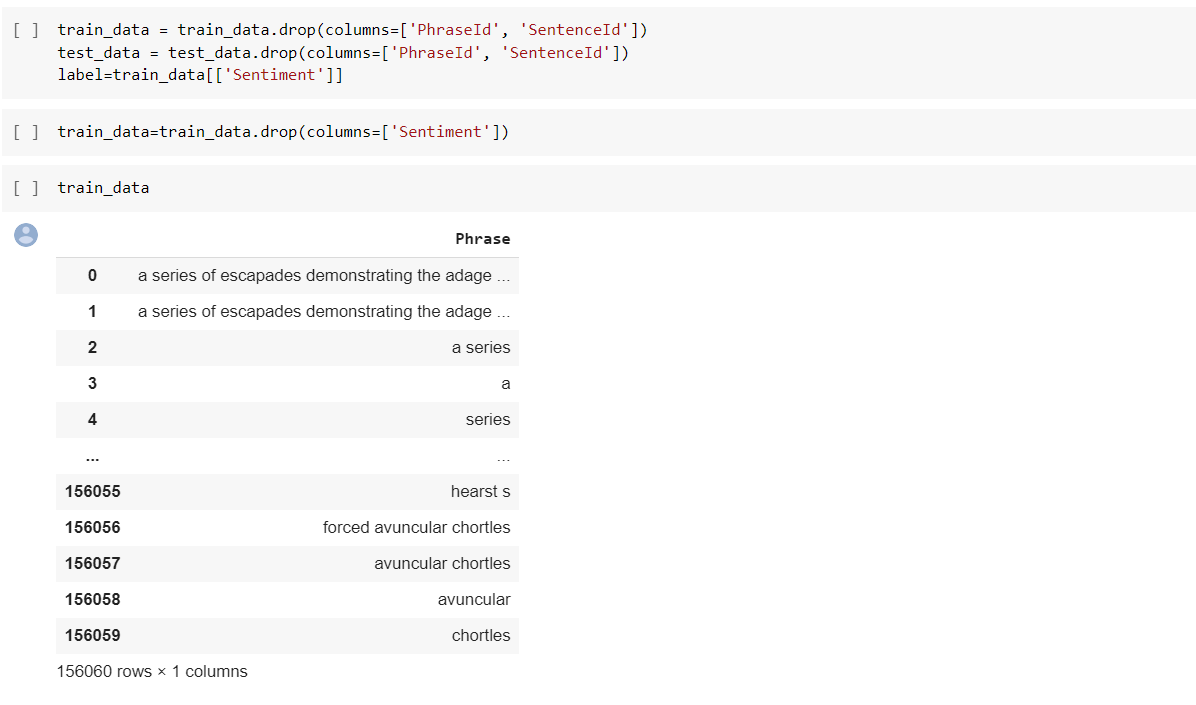




Task 5:

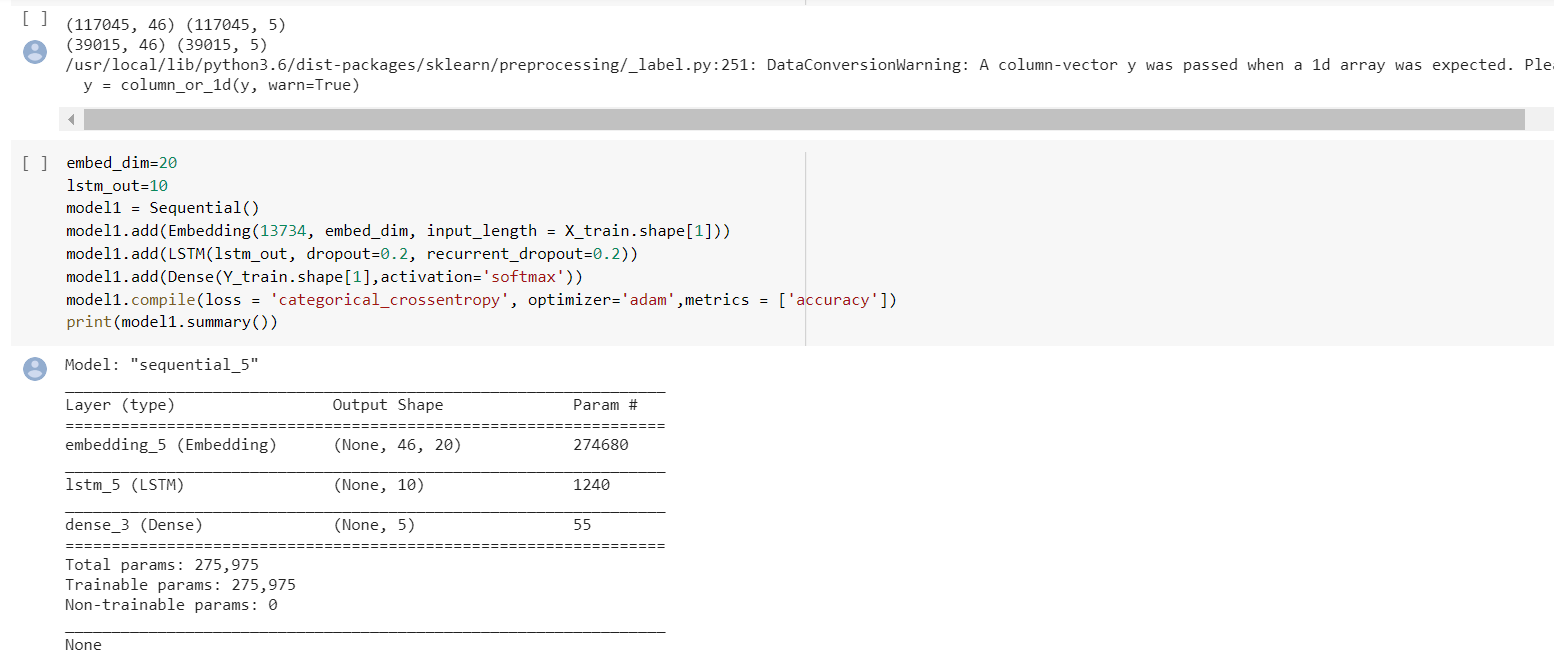
We used LSTM here.

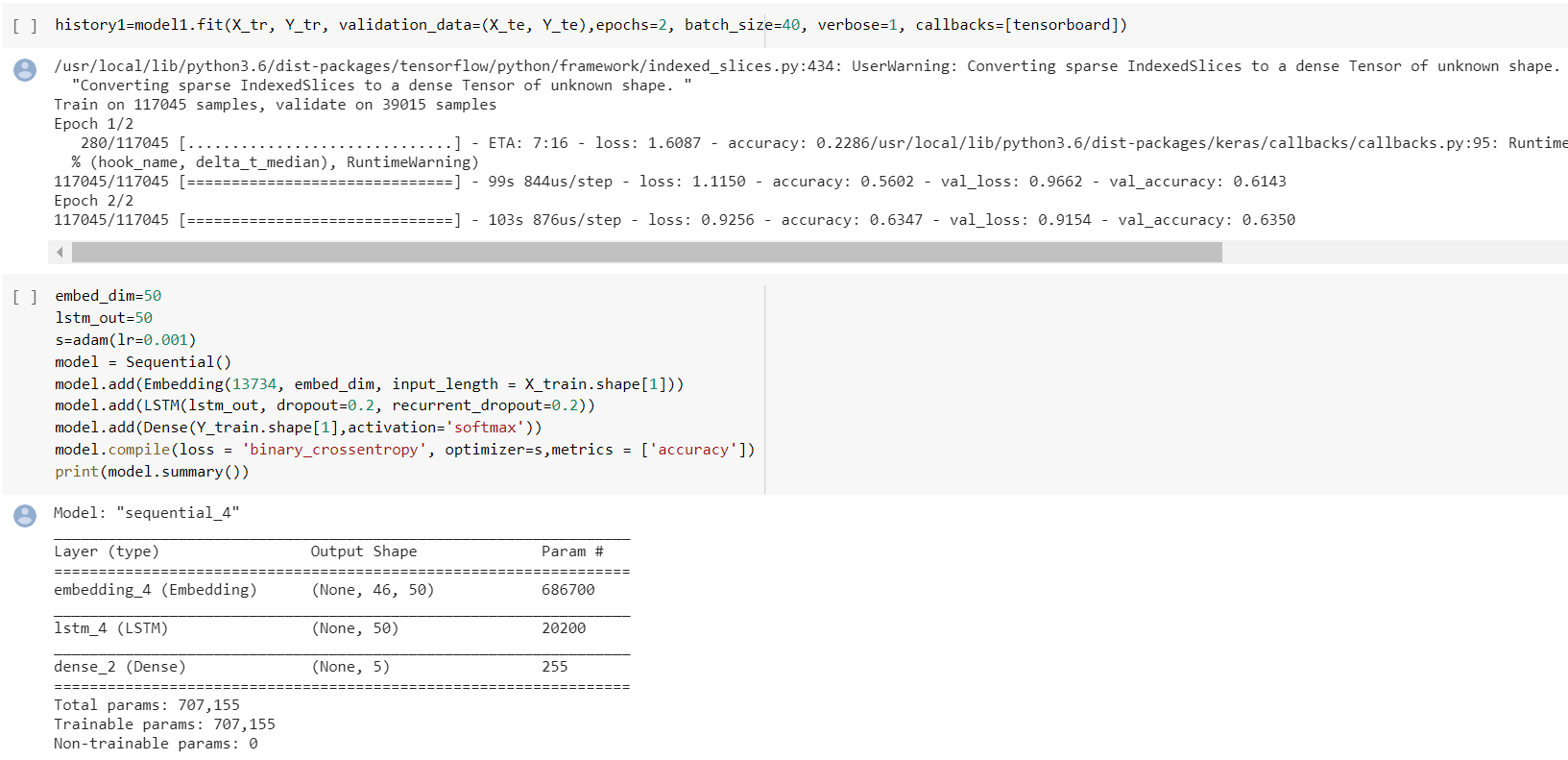




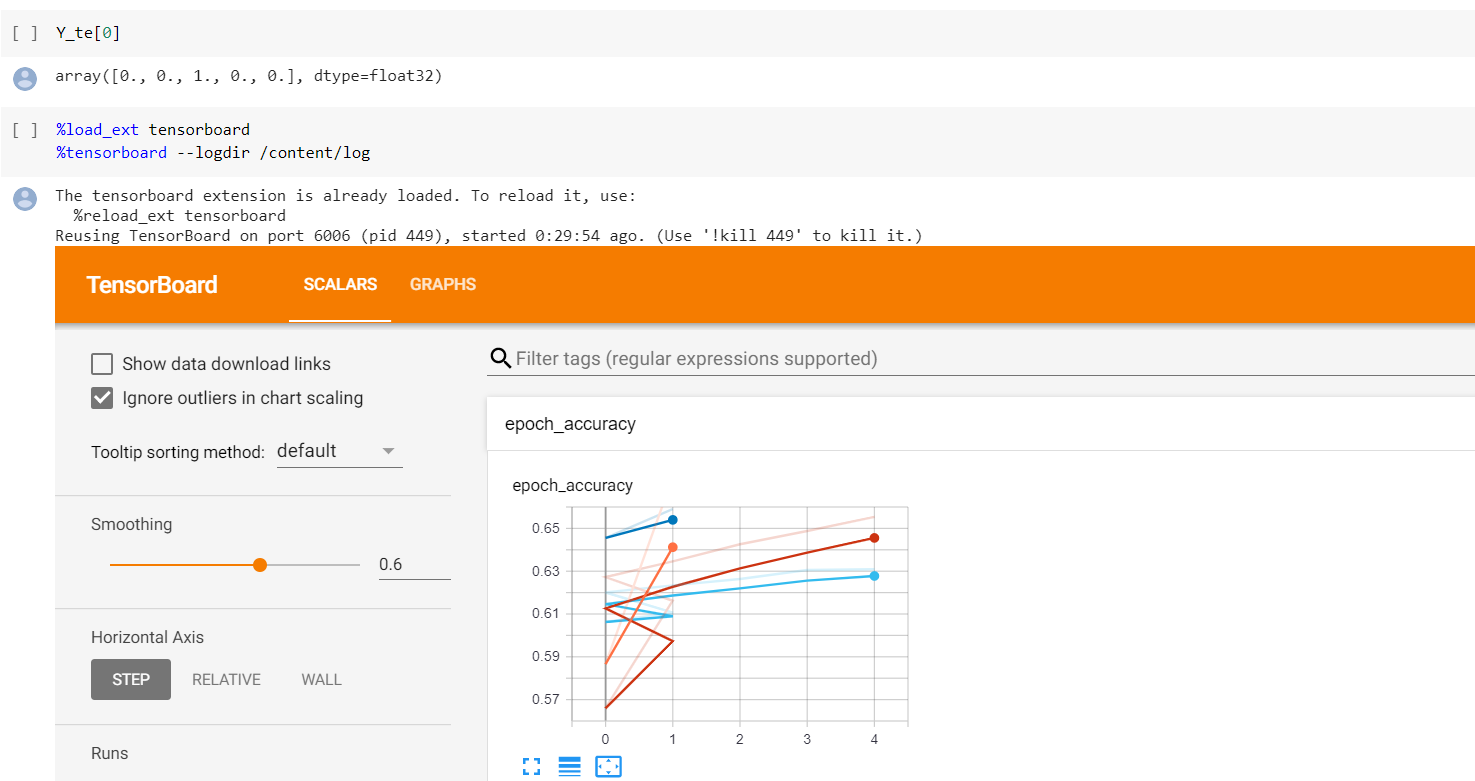


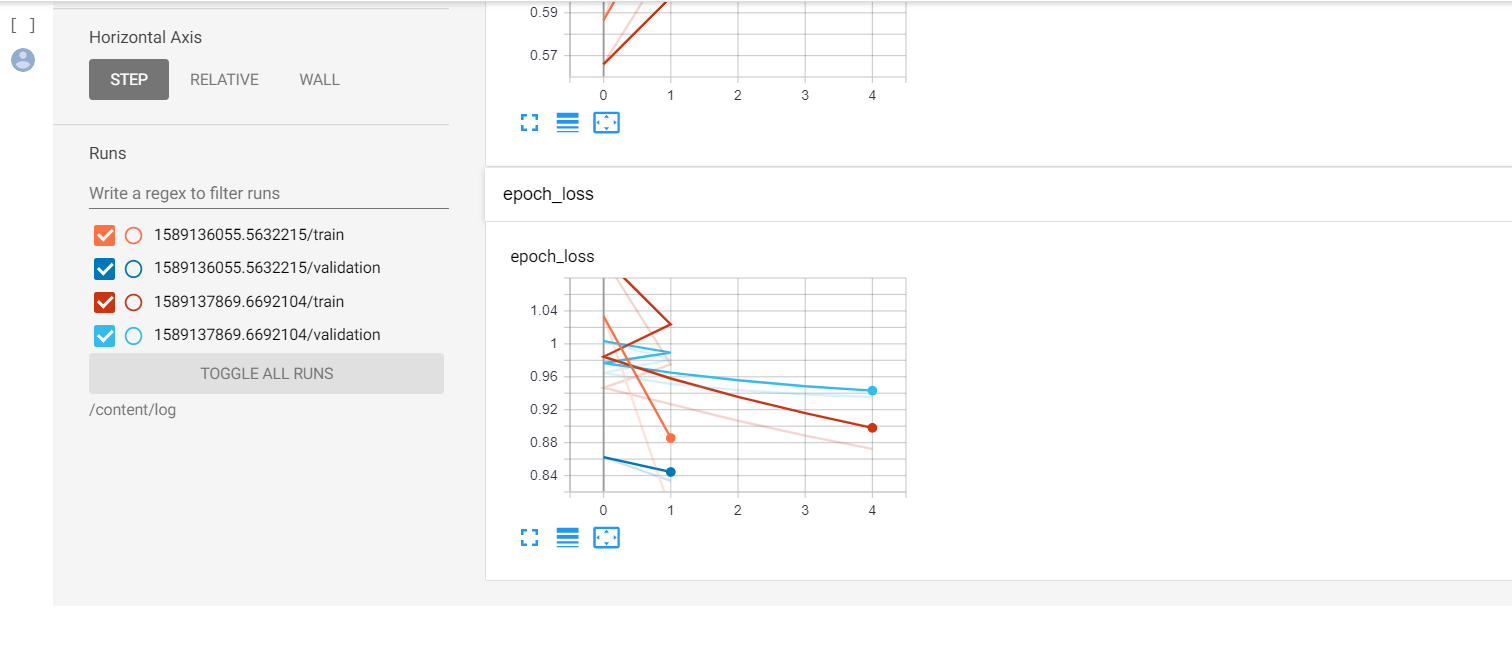












Task 6:

We can see from the above two models that accuracy is more for LSTM than CNN model. Then we have tuned our model.

We have altered the past projects for this execution utilizing CNN. We have taken the model of inquiry 4, and changed the hyper parameters. We have presented the learning rate, changed the quantity of neurons, and changed the misfortune capacity to binary\_crossentropy. While fitting the model we have changed the group size. We have acquired an exactness of 86%, which implies there is an expansion in the precision from 69% to 86%.

We have altered the past projects for this execution utilizing LSTM. We have taken the model of inquiry 5, and changed the hyper parameters. We have presented the learning rate, changed the quantity of neurons, and changed the misfortune capacity to binary\_crossentropy. While fitting the model we have changed the clump size. We have gotten a precision of 87%, which implies there is an expansion in the exactness from 63% to 87%.

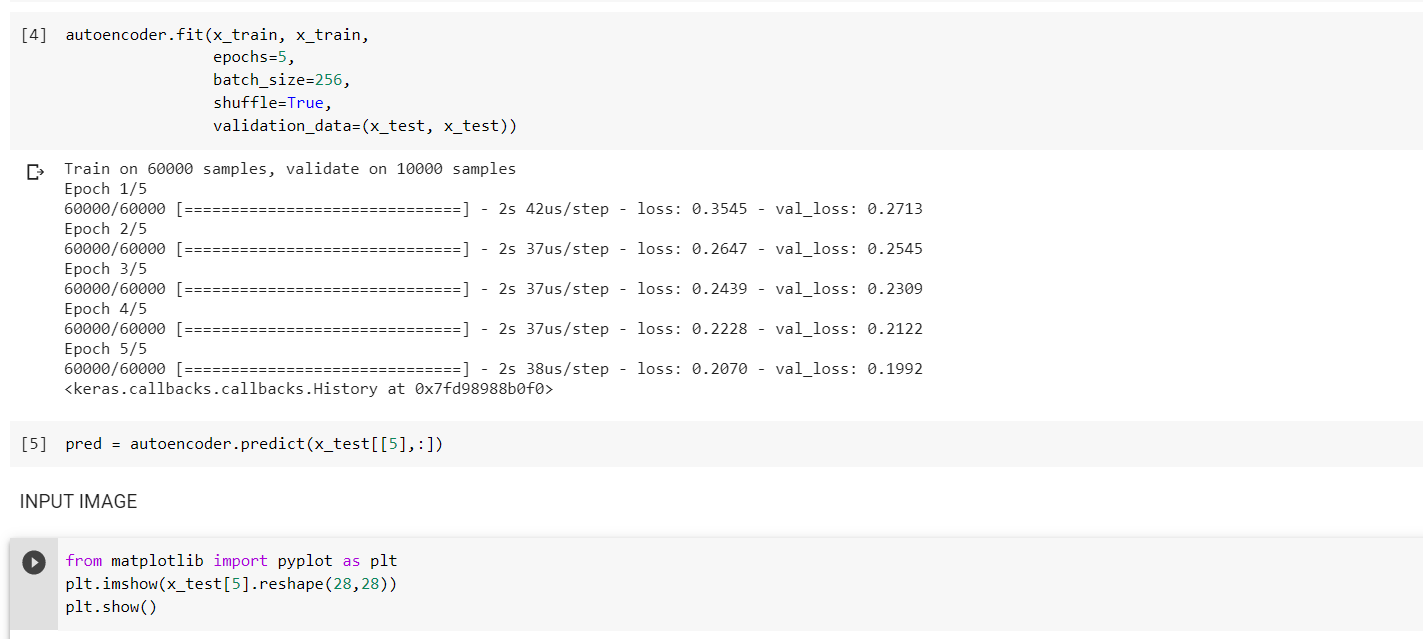
Task 7:

We have downloaded the MNIST dataset, and utilizing the autoencoders we have encoded and decoded a specific picture.

We have imported all the libraries that are required. We are stacking the information from mnist from keras. dataset library. Presently we are scaling the information by changing over it into glide and partitioning it by 255, so that navigating should be possible somewhere in the range of 0 and 1. At that point we are reshaping the preparation and testing information.



In this progression we perform encoding and deciphering and use autoencoder to manufacture the model and gather it utilizing adadelta as the streamlining agent and binary\_crossentropy as misfortune. Finally fit the model utilizing 15 ages and 256 as clump size. We use encoding to speak to the information and disentangling to get the lossy reproduction of the information.



As we were approached to play out the encoding and translating on a solitary picture. In this step we are showing the picture before encoding, which is from the test informational collection.



