Configuration Manual

MSc Research Project
Data Analytics

Mansi Atul Chowkkar
Student ID: x18134599

School of Computing
National College of Ireland

Supervisor:
Cristina Muntean

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Mansi Chowkkar
Student ID:	X18134599
Programme:	MSc. DATA ANALYTICS **Year:** 2019-2020
Module:	MSc. RESEARCH PROJECT
Lecturer:	Cristina Muntean
Submission Due Date:	**12/12/2019**
Project Title:	Breast Cancer Detection from Histopathological Images using Deep Learning and Transfer Learning.
Word Count:	721 **Page Count: 13**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ………

Date: ………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

1 Hardware/Software Requirements

1.1 Hardware Requirements

The below hardware would be ideal for running the experiment smoothly.

OS	Windows 10
RAM	Minimum 8GB (2.14 GB from Colaboratory)
Hard Disk Space	Minimum 100GB (100 GB drive space)

1.2 Software Requirements

Programming Language Tools	Google Colaboratory (Cloud based Jupyter notebook environment), Python version 3, Microsoft Excel, Overleaf
Web Browser	Google Chrome or Mozilla Firefox
Email	Access to a Gmail account

1 Google Colaboratory Environment Setup

This section will explain how to setup Google Colaboratory environment in order to perform this experiment. Screenshots are included in order to facilitate a better understanding. A new Gmail address as a mansithesis.unet@gmail.com account is created in order to access Google Colaboratory.

1. Sign in with your Gmail account:

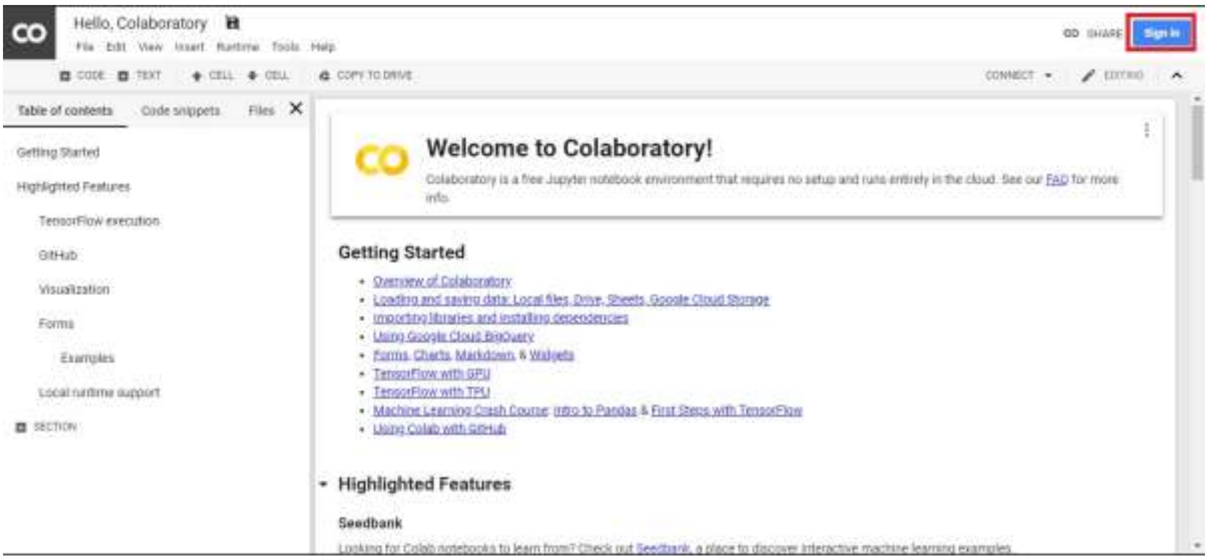

Figure 1: Sign in to Google Colaboratory

2. Once in the notebook, all required libraries are imported. The libraries required for each algorithm are written at the start of the coding of the algorithm.

2 Data preparation for Experiments

This section will explain how to upload data on Google Drive and access it in Collaboratory notebook.

1. Upload the image data folder on Google Drive:
 Histopathological image dataset of the Breast Biopsy is taken from the Kaggle as shown in the figure 2. Images of all magnifications are mixed in one folder for executing Experiment 1 and image data folder is uploaded on Google drive as shown in figure 3.

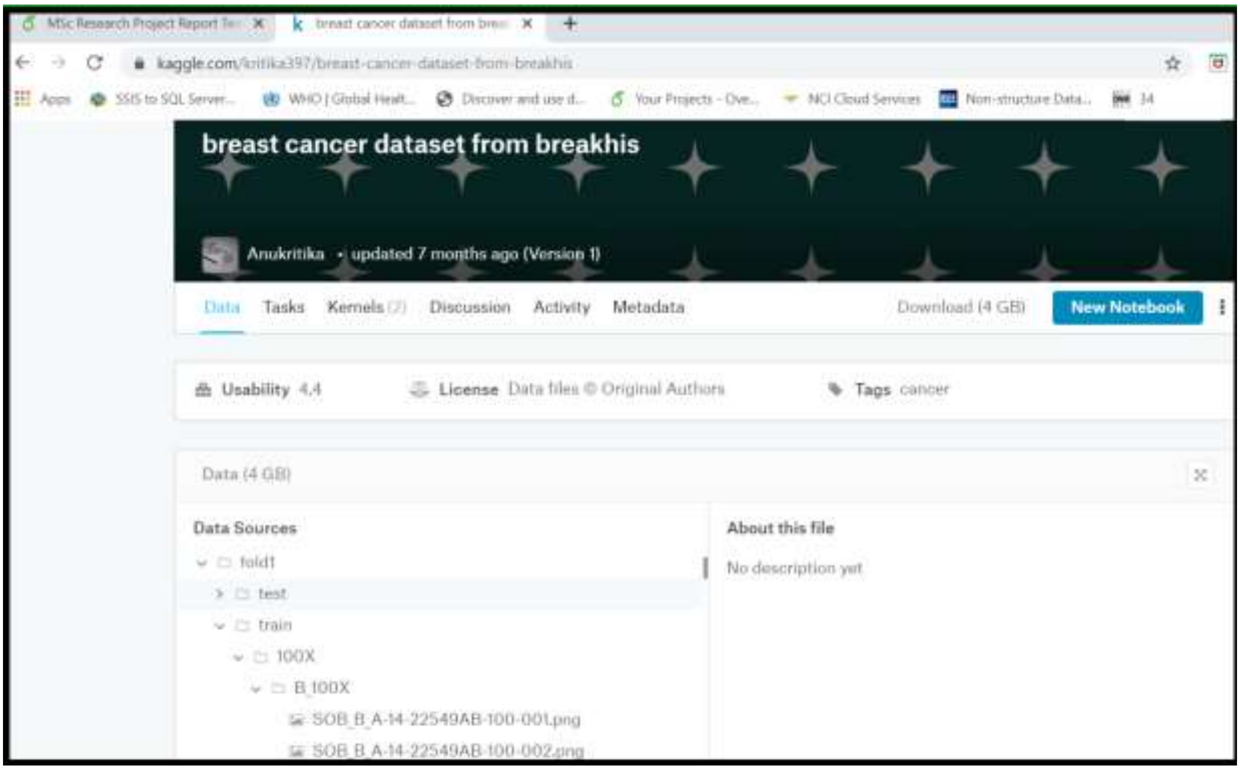

Figure 2:DataSet from Kaggle

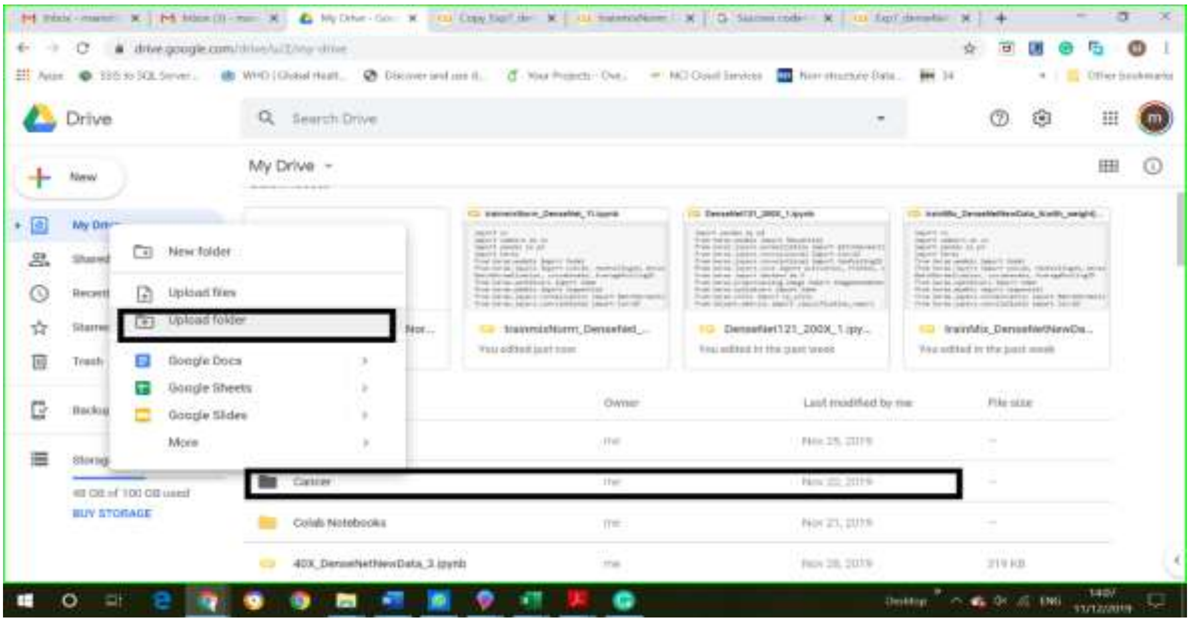

Figure 3: Upload data on Drive

2. Mount Google drive in the colaboratory notebook:

Figure 4 shows the steps for mounting drive in the python code. Click on the url and select the gmail account for the colaboratory and enter the authorization code as shown in figure 5.

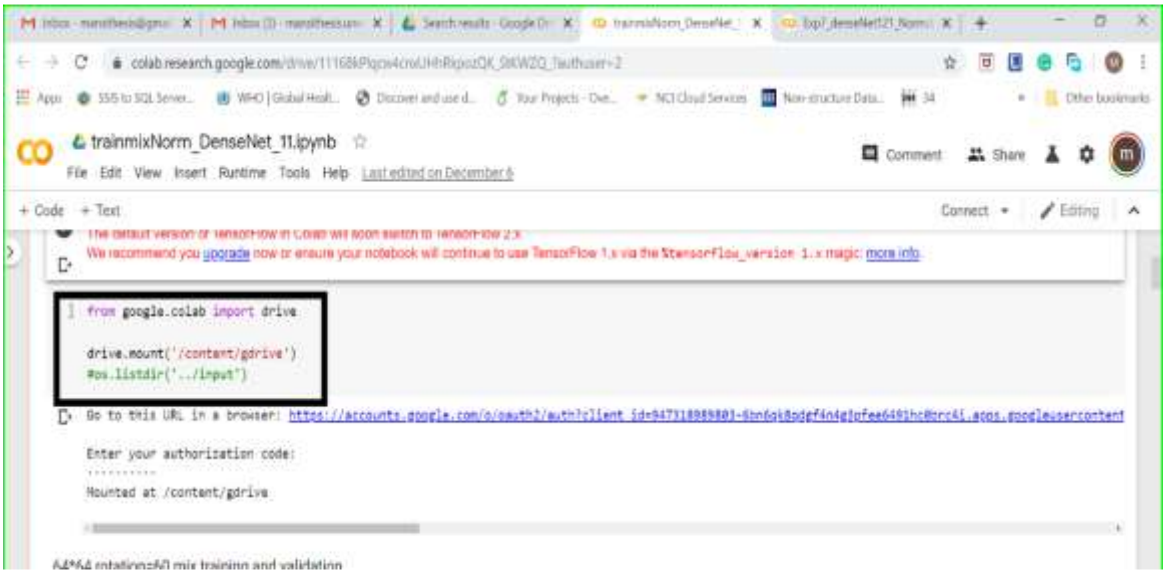

Figure 4: Mounting of Google Drive

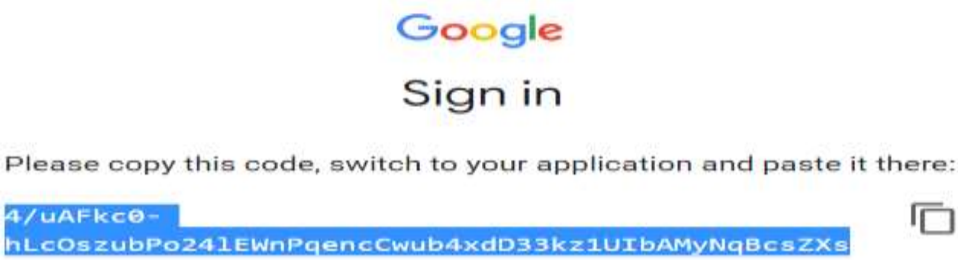

Figure 5: Enter Authorization code

3. Access the image data using below command in the notebook:

Figure 6 shows the code for accessing images from folder present on google drive.

Figure 6: Accessing images in the python notebook

4. Import Libraries for image normalization and augmentation:

   Libraries required for Histogram normalisation are imported in python notebook as shown in figure7. ImageDataGenerator is used from Keras library to augment the images. Images are saved in the same directory for further use. Figure 7 shows the libraries required for the upsampling method.

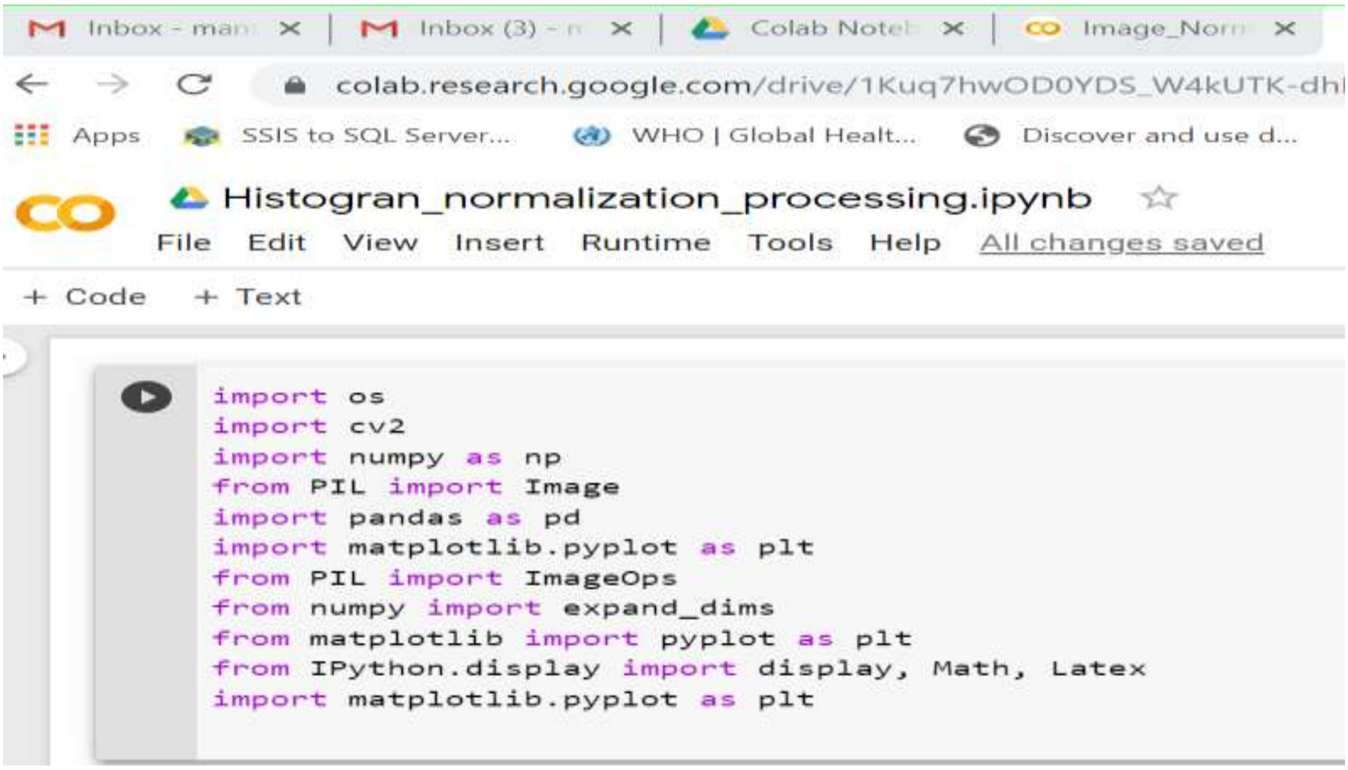

Figure 7: Libraries for Histogram Normalization

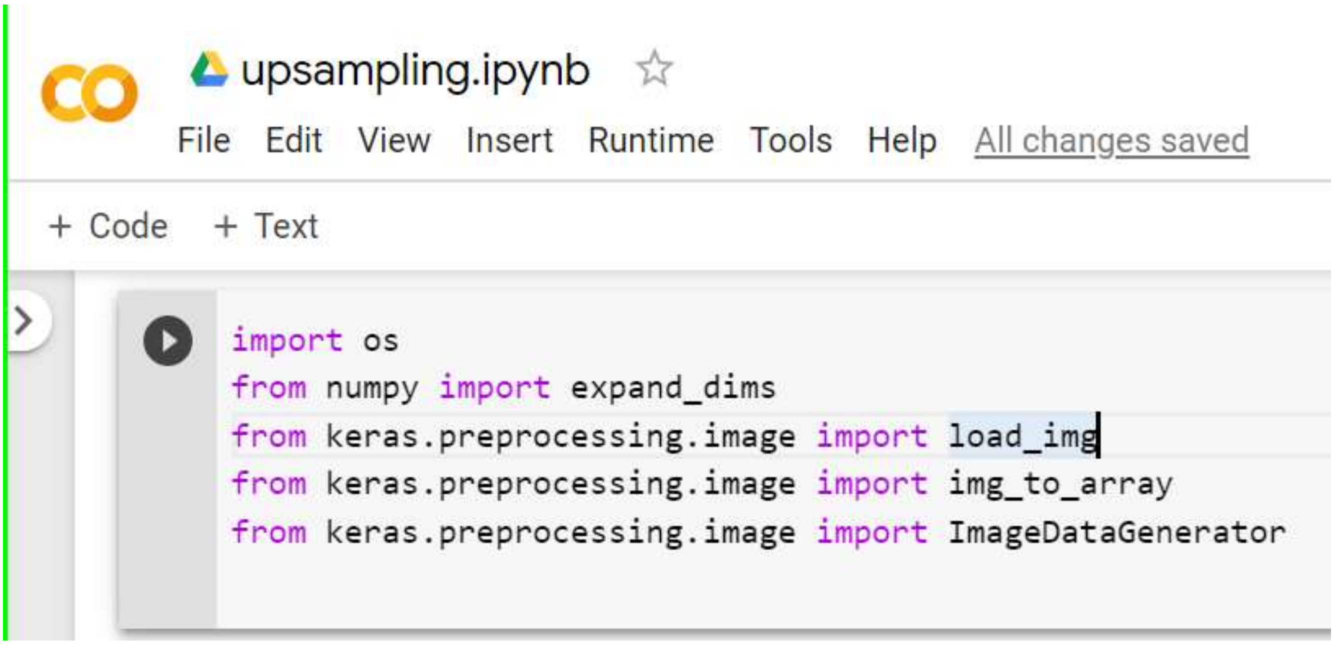

Figure 8: Image Augmentation

3 Deep Learning and Transfer Learning Model Execution

This section will explain which libraries need to be imported for the execution of CNN and DenseNet-121

1. CNN Execution:

For CNN execution keras and tensorflow libraries are used from python. Figure 9 shows all the required libraries for image augmentation, model definitions, model compiling and model execution. Also, libraries are required for statistical analysis and graph plotting. The random seed is set every time so that every time different output is generated for different experiments.

- All layers in the CNN model are imported from keras.layers libraries.
- The model is built using tensorow as tf library.
- The model layers are plotted using tensorow.keras.utils library

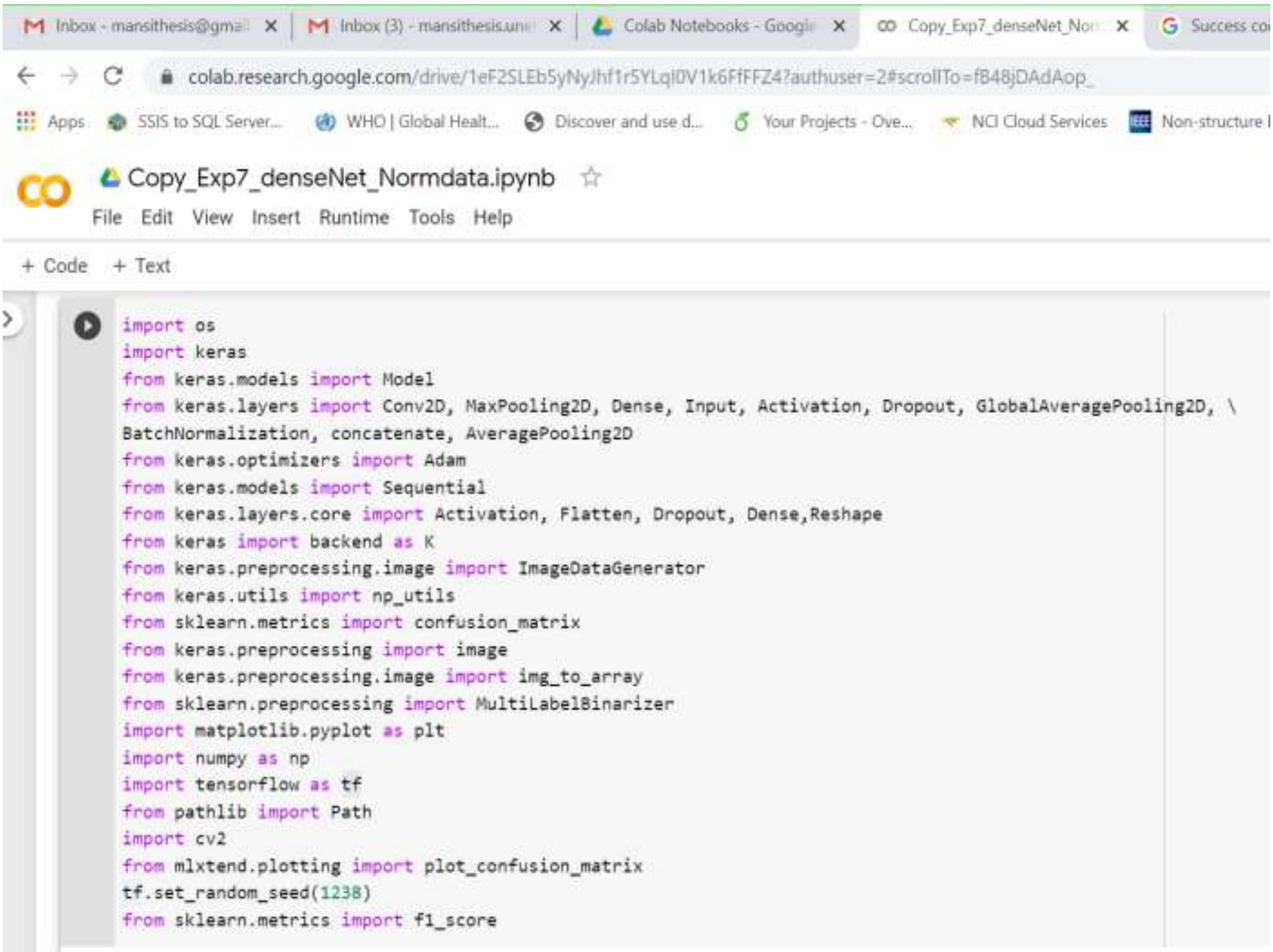

Figure 9: Libraries for CNN execution

2. DenseNet-121 Execution:

- For DenseNet-121 execution keras and tensorflow libraries are used from python. Figure 10 shows all the required libraries for image augmentation, model definitions, model compiling and model execution. Also, libraries are required for statistical analysis and graph plotting.
- DenseNet121 library is imported from keras package.
- Plotting of testing and training plots: matplotlib.pyplot as plt, seaborn as sns libraries are used for plotting training and validation graphs.

Copy of DenseNet121_400X_1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
[6] import os
    from keras.layers.normalization import BatchNormalization
    from keras.layers.convolutional import Conv2D, MaxPooling2D
    from keras.layers.core import Activation, Flatten, Dropout, Dense,Reshape
    from keras import backend as K
    from keras.preprocessing.image import ImageDataGenerator
    from keras.optimizers import Adam
    from keras.utils import np_utils
    from keras.preprocessing import image
    from keras.preprocessing.image import img_to_array
    from sklearn.preprocessing import MultiLabelBinarizer
    import matplotlib.pyplot as plt
    import numpy as np
    import argparse
    import tensorflow as tf
    import cv2
    from mlxtend.plotting import plot_confusion_matrix
    from keras.applications import densenet
    from keras.models import Sequential, Model, load_model
    import seaborn as sn
    from sklearn.metrics import confusion_matrix, classification_report
    tf.set_random_seed(1234)
    from keras import layers
    from keras.applications import DenseNet121
```

Figure 10: Libraries for execution of DenseNet-121

4 Settings done for accelerating Computation time

This section will explain about how the drive storage is extended and GPU setting is done from the google colaboratory.

1. Drive Storage:

Extra 100 GB drive storage is subscribed to store normalized and augmented images. Extra storage was required for storing different python files with the executed models. Figure 11 shows the utilization of google drive for this project.

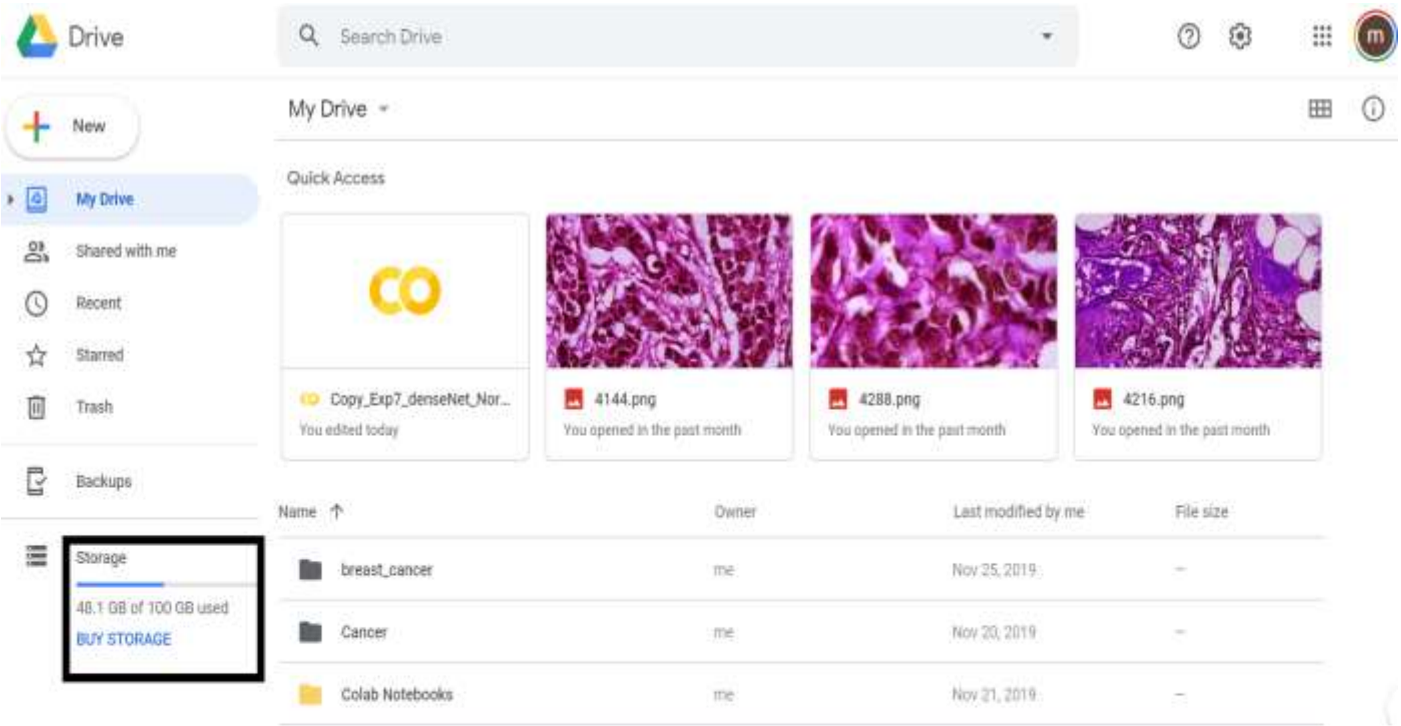

Figure 11: Storage

2. GPU:

For boosting the execution speed for both the models GPU accelerator hardware is selected from runtime option as shown in figure 12 and figure 13.

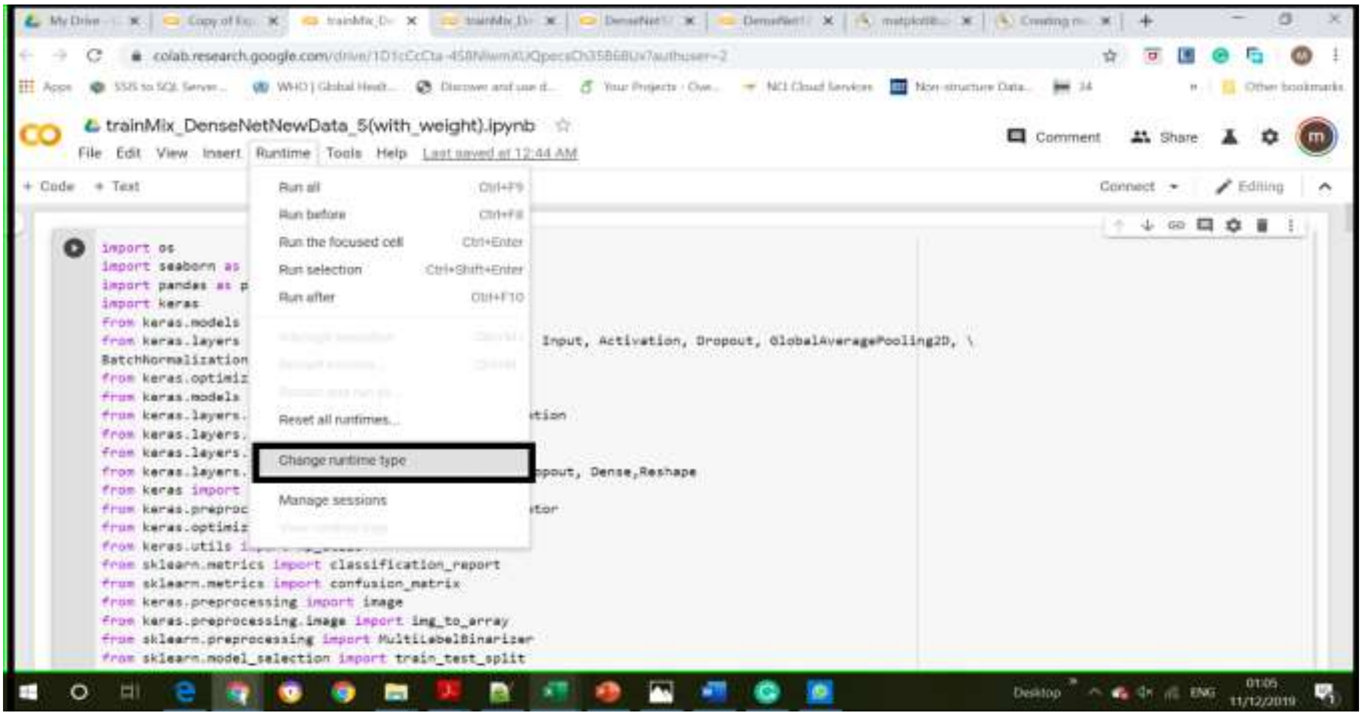

Figure 12: Runtime selection of hardware

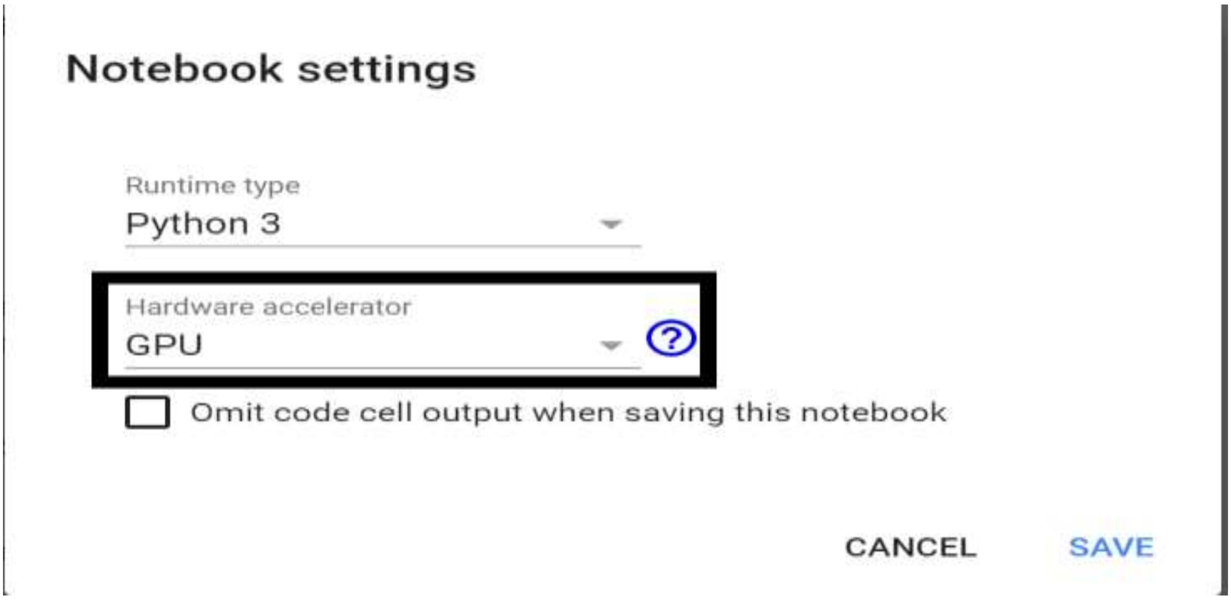

Figure 13: GPU selection

5. Other Software used

For visualization of evaluation results Microsoft Excel is used. Research report is written using Overleaf tool. Figure 14 shows the utilization of Overleaf for this project.

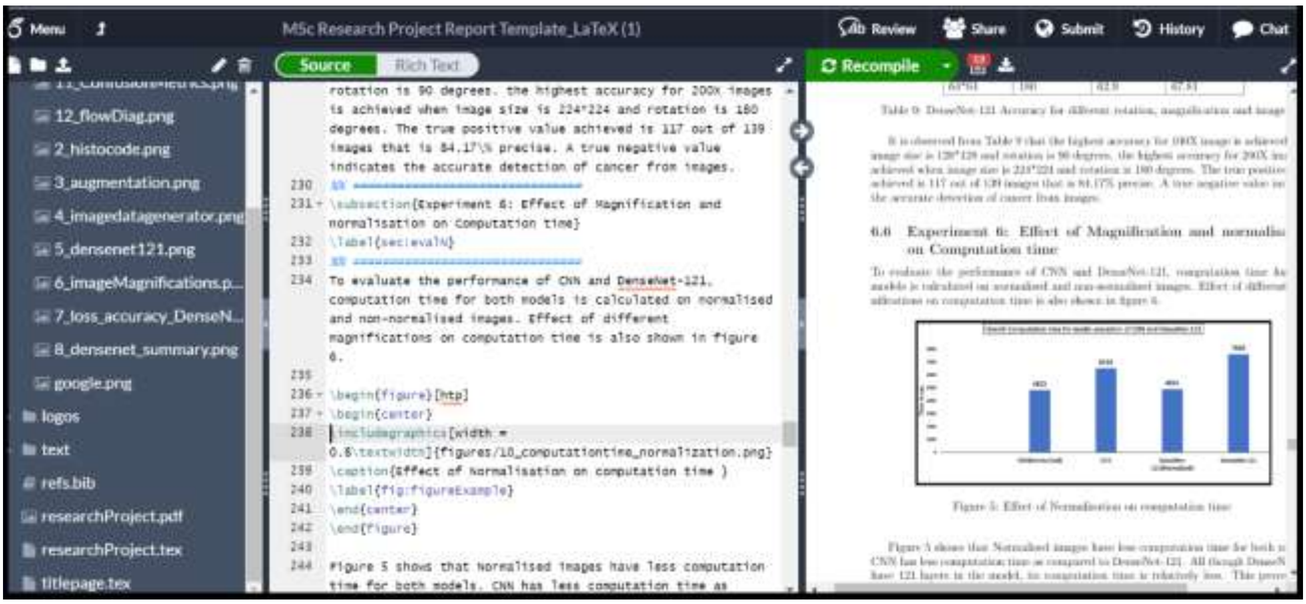

Figure 14: Overleaf Project

References

https://keras.io/preprocessing/image/

https://medium.com/hugoferreiras-blog/confusionmatrixandothermetricsinmachinelearning-894688cb1c0a

https://keras.io/getting-started/sequential-model-guide/

https://matplotlib.org/3.1.1/tutorials/introductory/pyplot.html