

Assignment 3

Question 1

- Import required libraries and write device agnostic code
- Load dataset and make dataset class for take selected 8 attribute from dataset
- Make dataloader of 32 batchsize
- **For 8 attribute i preferred to choose** ['Attractive', 'Bald', 'Big_Nose', 'Black_Hair', 'Blond_Hair', 'Eyeglasses', 'Mustache', 'No_Beard']
- **Because of they are easily detect in low resolution of image**
- I take half of the data for training and tests due to GPU constraint in CoLab. Therefore, 75000 images are used for trains and 25000 images are used for tests.
- Now write the train and test loop for training and testing.
- Take cross entropy as loss function and take third party accuracy function from torch metric.
- Write plot function for Loss/Accuracy vs Epoches
- Now build a multitask model using resnet18.
- Now train model for 5 epochs.

Result:

Epoch: 1

Train Accuracy

attribute 1: 0.7708 attribute 2: 0.5947 attribute 3: 0.7297 attribute 4:
0.8094
attribute 5: 0.8730 attribute 6: 0.8492 attribute 7: 0.5340 attribute 8:
0.8416

Test Accuracy

attribute 1: 0.7964 attribute 2: 0.5537 attribute 3: 0.7252 attribute 4:
0.8216
attribute 5: 0.9135 attribute 6: 0.8841 attribute 7: 0.6233 attribute 8:
0.8644

Train Loss: 1.8865 | Test Loss: 1.6711 | Train Accuracy: 0.7503 | Test
Accuracy: 0.7728

Epoch: 2

Train Accuracy

attribute 1: 0.7930 attribute 2: 0.6503 attribute 3: 0.7565 attribute 4:
0.8353
attribute 5: 0.8922 attribute 6: 0.8890 attribute 7: 0.5784 attribute 8:
0.8818

Test Accuracy

attribute 1: 0.7912 attribute 2: 0.6787 attribute 3: 0.7401 attribute 4:
0.8259
attribute 5: 0.9068 attribute 6: 0.8884 attribute 7: 0.6498 attribute 8:
0.8828

Train Loss: 1.6346 | Test Loss: 1.6293 | Train Accuracy: 0.7846 | Test
Accuracy: 0.7955

Epoch: 3

Train Accuracy

attribute 1: 0.7989 attribute 2: 0.6695 attribute 3: 0.7684 attribute 4:
0.8461
attribute 5: 0.9005 attribute 6: 0.9055 attribute 7: 0.6159 attribute 8:
0.8927

Test Accuracy

attribute 1: 0.8033 attribute 2: 0.6328 attribute 3: 0.7687 attribute 4:
0.8772
attribute 5: 0.9016 attribute 6: 0.9033 attribute 7: 0.5511 attribute 8:
0.8862

Train Loss: 1.5229 | Test Loss: 1.5299 | Train Accuracy: 0.7997 | Test
Accuracy: 0.7905

Epoch: 4

Train Accuracy

attribute 1: 0.8061 attribute 2: 0.6732 attribute 3: 0.7753 attribute 4:
0.8552
attribute 5: 0.9068 attribute 6: 0.9111 attribute 7: 0.6473 attribute 8:
0.9003

Test Accuracy

attribute 1: 0.8088 attribute 2: 0.6723 attribute 3: 0.7642 attribute 4:
0.8669
attribute 5: 0.9103 attribute 6: 0.9197 attribute 7: 0.6369 attribute 8:
0.8933

Train Loss: 1.4440 | Test Loss: 1.4835 | Train Accuracy: 0.8094 | Test
Accuracy: 0.8090

Epoch: 5

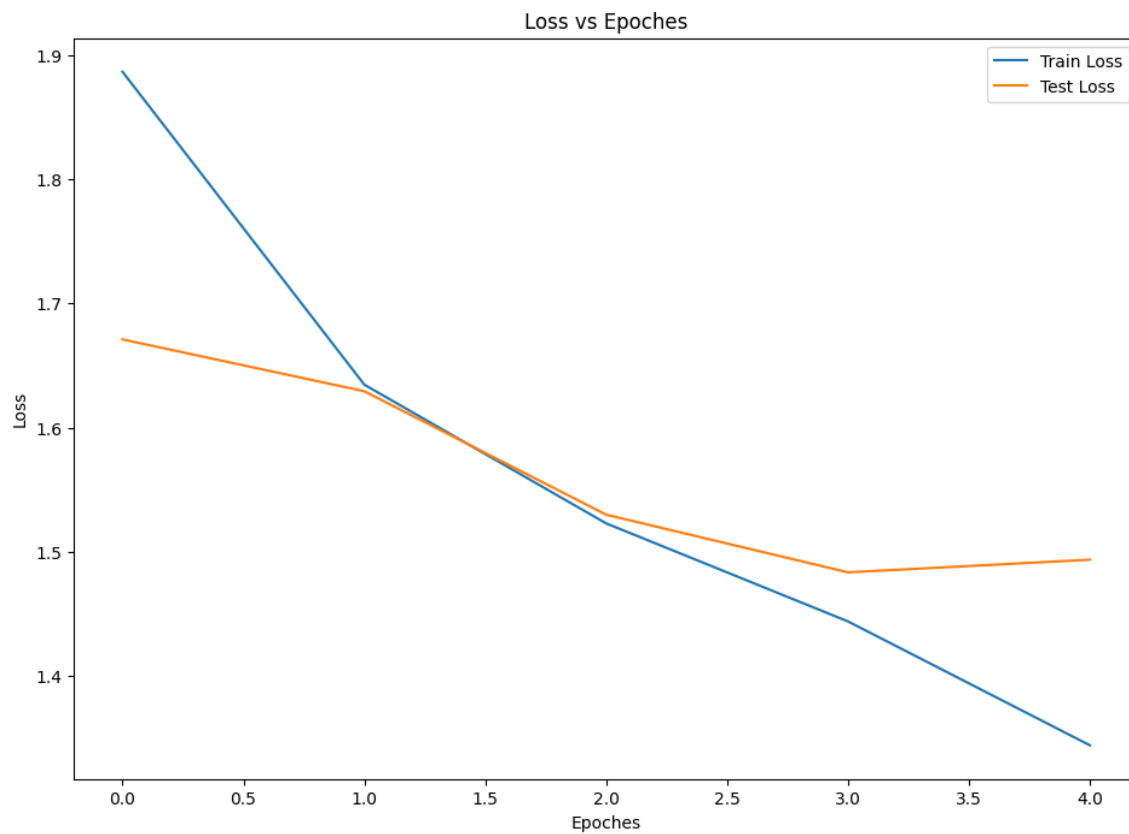
Train Accuracy

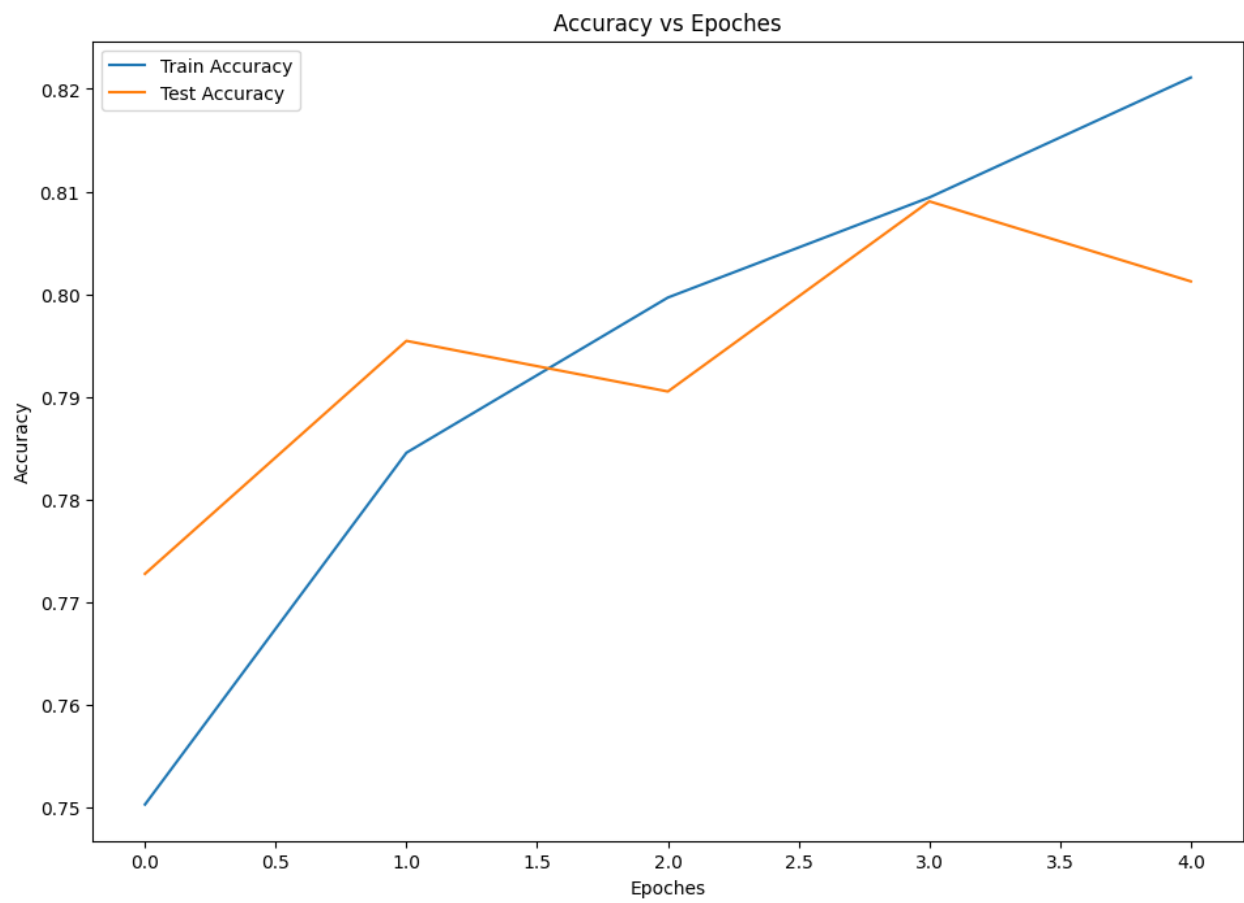
attribute 1: 0.8173 attribute 2: 0.6848 attribute 3: 0.7883 attribute 4:
0.8650
attribute 5: 0.9129 attribute 6: 0.9095 attribute 7: 0.6772 attribute 8:
0.9137

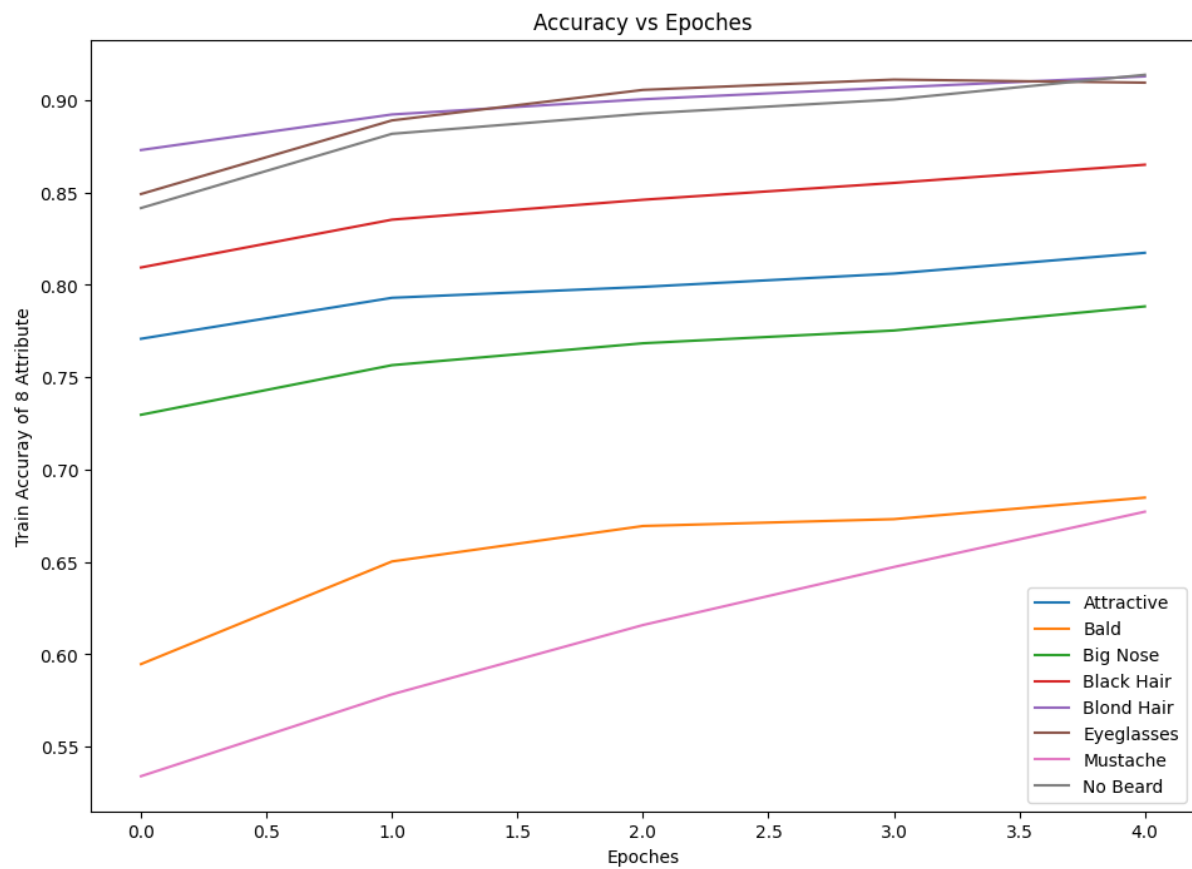
Test Accuracy

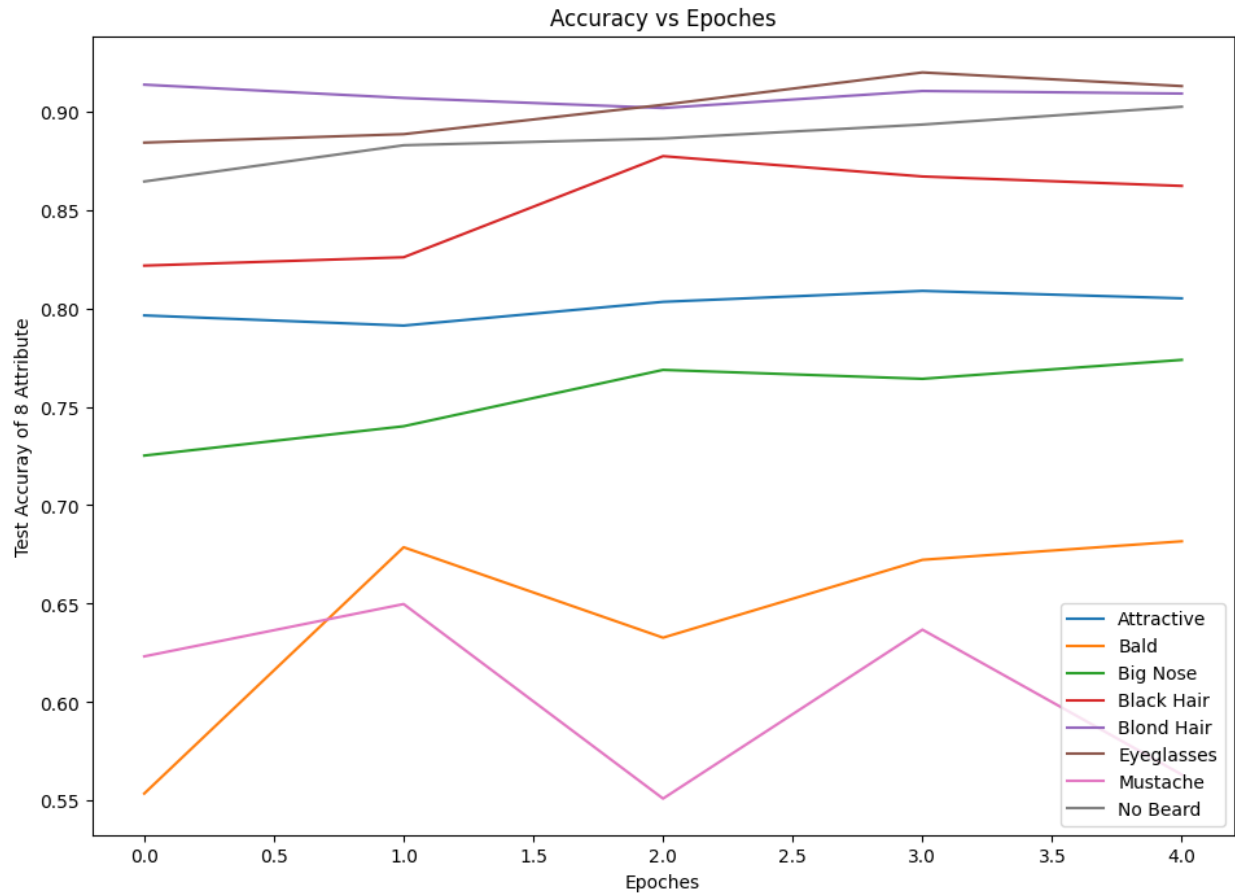
attribute 1: 0.8050 attribute 2: 0.6817 attribute 3: 0.7738 attribute 4:
0.8621
attribute 5: 0.9090 attribute 6: 0.9128 attribute 7: 0.5632 attribute 8:
0.9023

Train Loss: 1.3443 | Test Loss: 1.4937 | Train Accuracy: 0.8211 | Test
Accuracy: 0.8012









Observation:

- Overall model performs well in just 5 epochs.
- Overall accuracy of model is good and if we run for it more epoch till 15 we get overall accuracy up to 90
- In 8 individual attributes we get poor performance in Mustache attribute and Bald.
- On the other hand, I got the best performance in No Beard, Eyeglasses, Blond Hair and Black Hair more than 80 percent.
- However, multi-tasking model take too much time to train.