COMP-5011-FDE Machine Learning & Neural Network

Assignment 5

Name: Khushal Paresh Thaker

Student ID: 1106937

Conditions

- (the last digit of ID belongs to 5-7): CIFAR10
- VGG16 (the second last digit of student ID belongs to 3-4)

Content:

Part 1: Transfer Learning Model

First Run

Training Accuracy: 0.9526000022888184
Training Loss: 0.183
Validation Accuracy: 0.9746
Validation Loss: 0.0977
Test loss: 0.7521984577178955
Test accuracy: 0.8572999835014343
Training Execution Time (15 epoch): 509.673 seconds

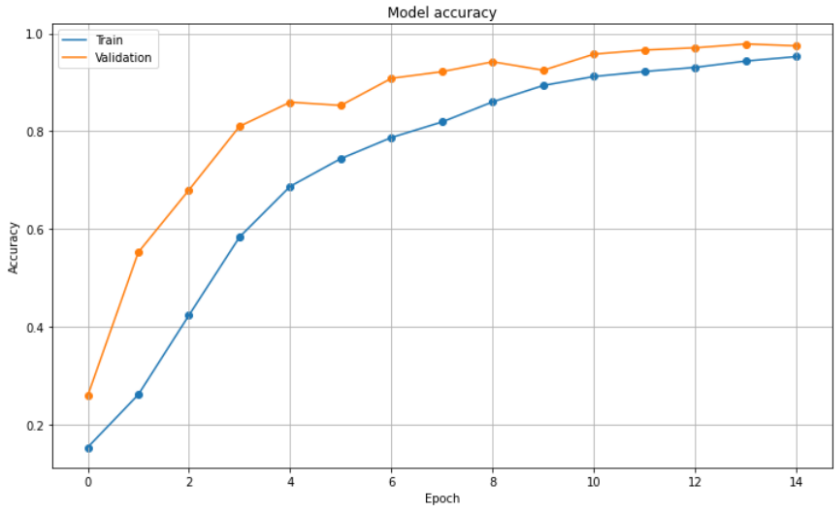

Fig 1: Accuracy vs Epoch for First Run - Transfer Learning Model

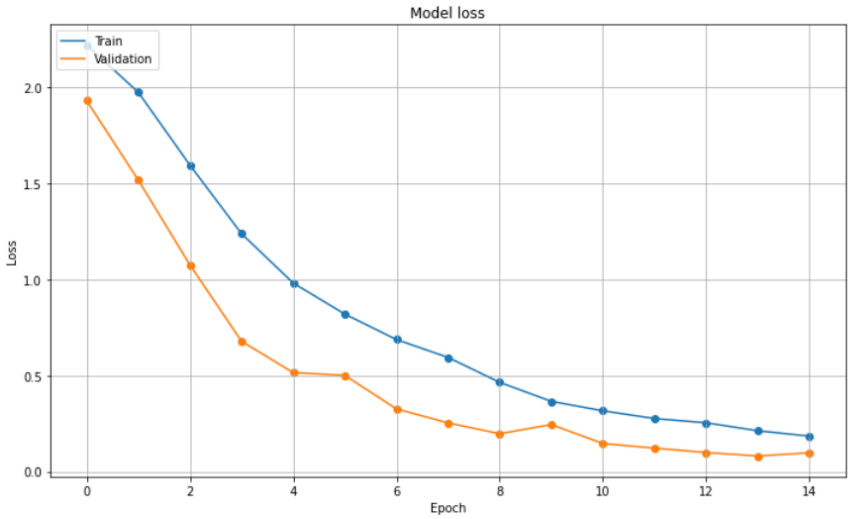

Fig 2: Loss vs Epoch for First Run - Transfer Learning Model

Second Run

Training Accuracy: 0.9676399827003479
Training Loss: 0.1236
Validation Accuracy: 0.9844
Validation Loss: 0.0452
Test loss: 0.9187352657318115
Test accuracy: 0.8600000143051147
Training Execution Time (15 epoch): 949.784 seconds

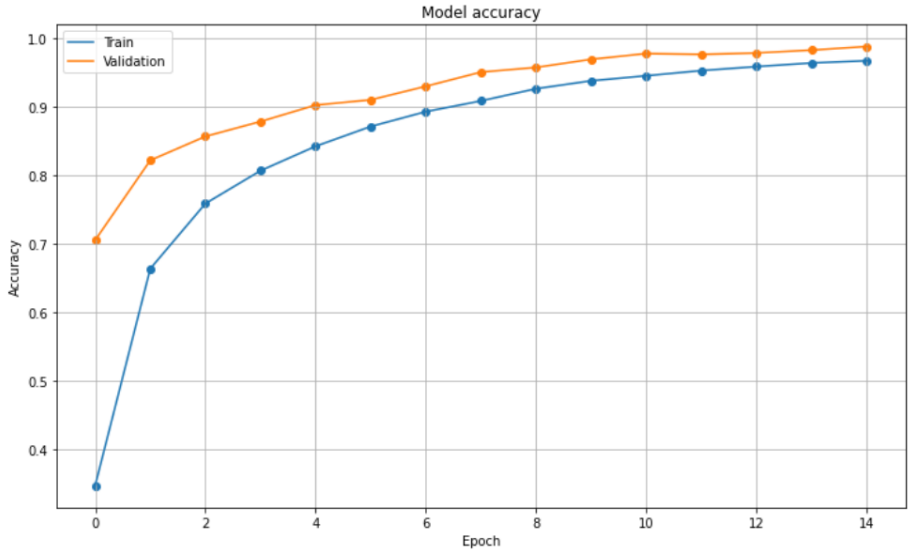

Fig 3: Accuracy vs Epoch for Second Run - Transfer Learning Model

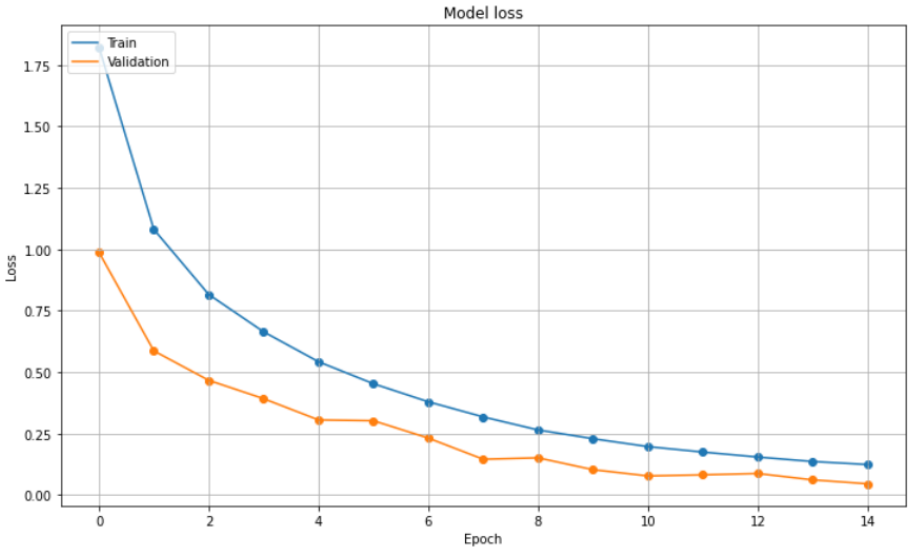

Fig 4: Loss vs Epoch for Second Run - Transfer Learning Model

Third Run

Training Accuracy: 0.9676399827003479
Training Loss: 0.1236
Validation Accuracy: 0.9844
Validation Loss: 0.0452
Test loss: 0.7465143799781799
Test accuracy: 0.8611000180244446
Training Execution Time (15 epoch): 1388.106 seconds

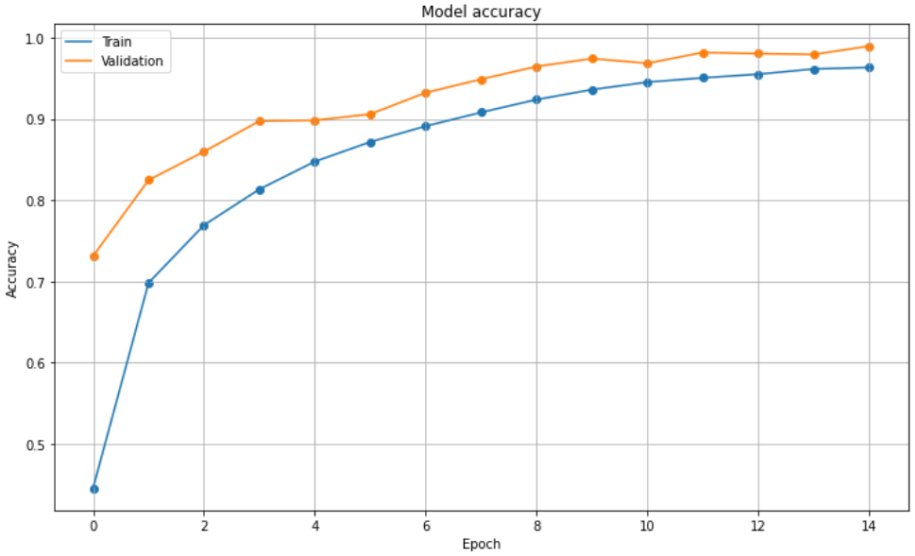

Fig 5: Accuracy vs Epoch for Third Run - Transfer Learning Model

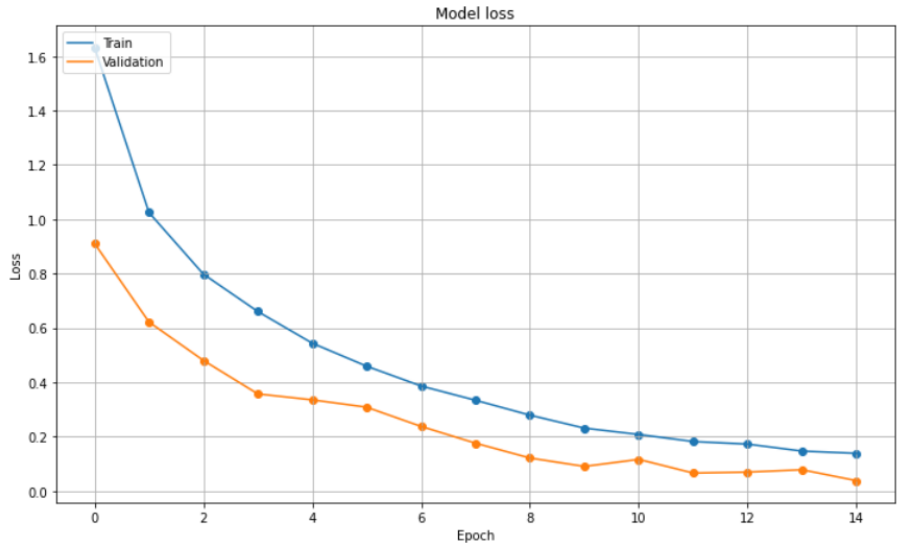

Fig 6: Loss vs Epoch for Third Run - Transfer Learning Model

Average Accuracies

Average of Training Accuracy for all three runs of transfer learning model: 96.1193323135376%

Average of Testing Accuracy for all three runs of transfer learning model: 85.947 %

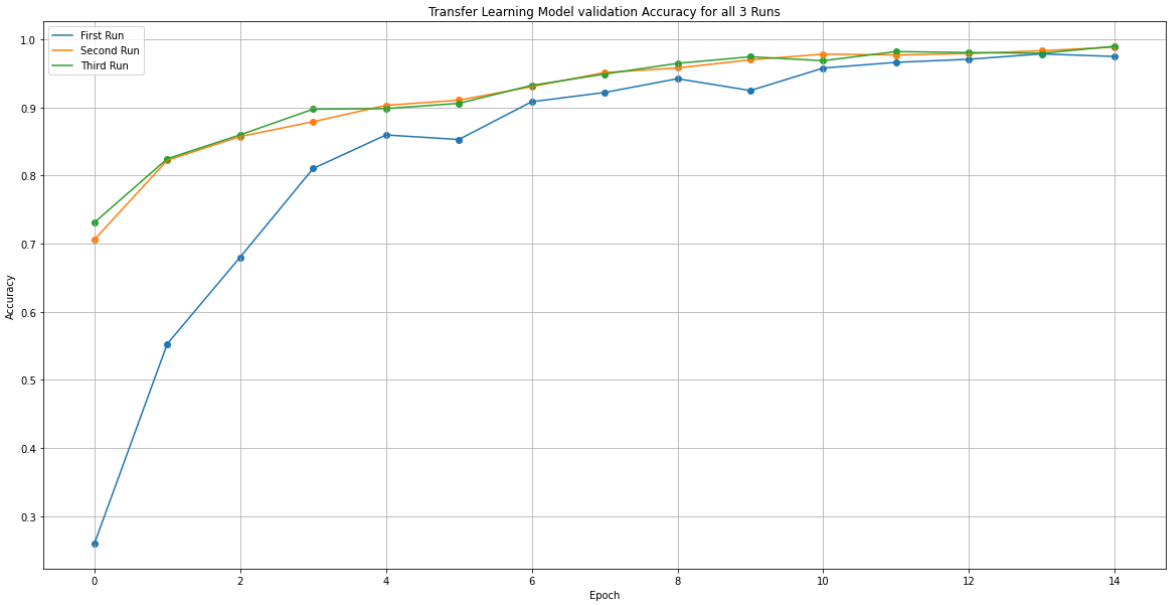

Fig 7: Training Accuracy vs Epoch for all three runs of transfer learning model

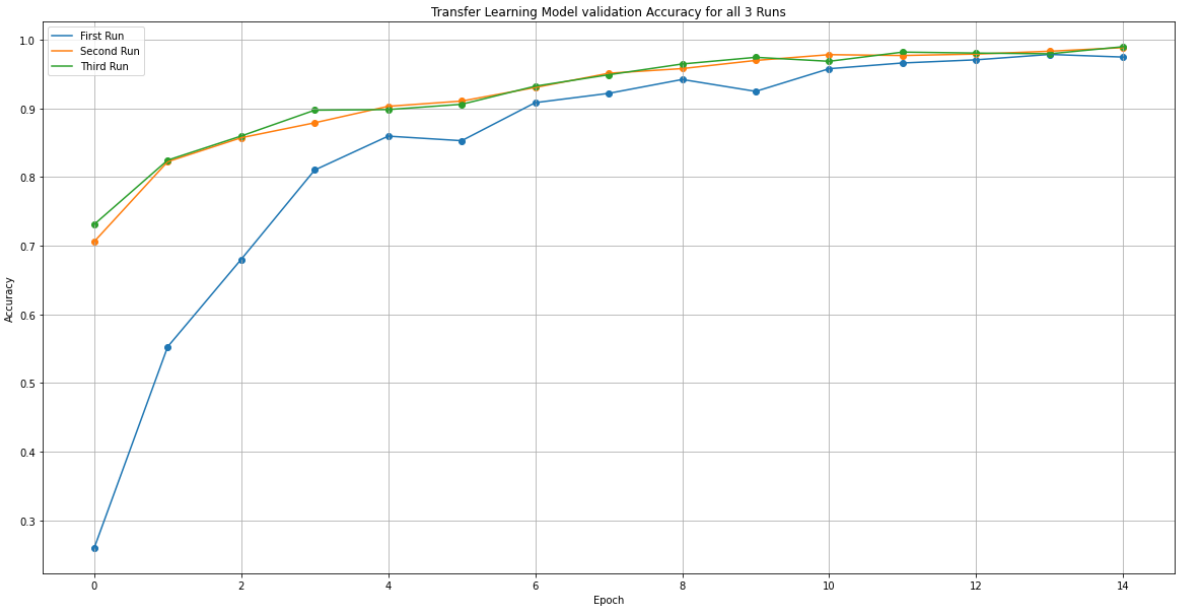

Fig 8: Validation Accuracy vs Epoch for all three runs of transfer learning model

Task 2: Learn from Scratch

Training Loss: 0.1843
Training Accuracy: 0.9494
Validation loss: 0.0607
Validation accuracy: 0.9823
Test loss: 0.5767048001289368
Test accuracy: 0.8655999898910522
Training Execution Time (125 epochs): 3175.482 seconds

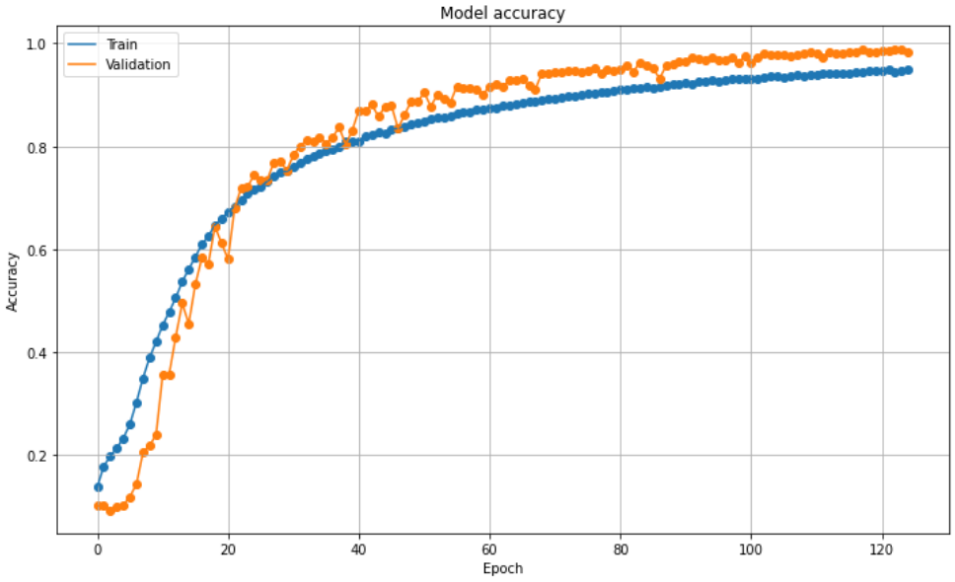

Fig 9: Accuracy vs Epoch for Learn from Scratch model

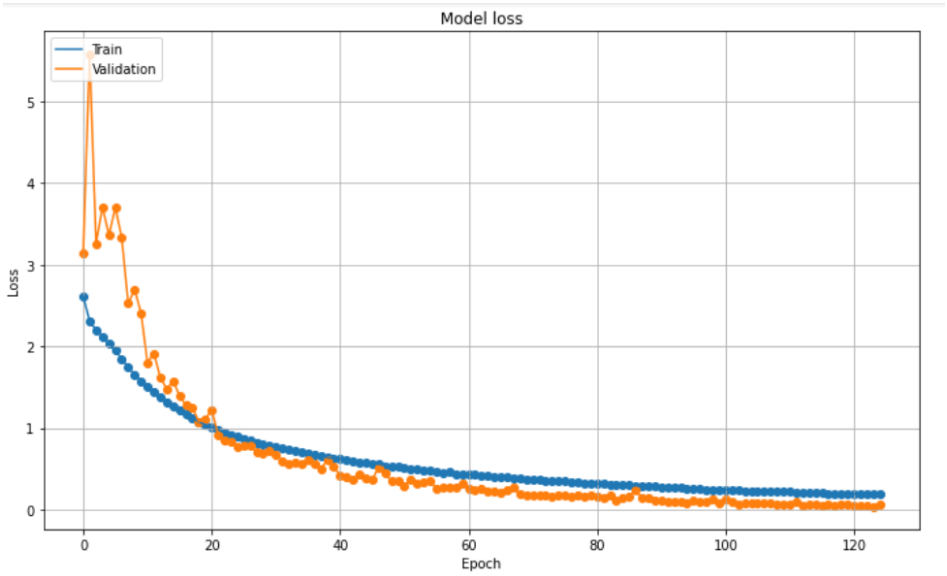

Fig 9: Loss vs Epoch for Learn from Scratch model

Part 3: Comparison of Performance

Accuracy Comparison

Transfer Learning Model
Average Training Accuracy of Transfer Learning Model for 3 runs: 96.1193323135376
Average Testing Accuracy of Transfer Learning Model for 3 runs: 85.94666719436646
Learn From Scratch Model
Training Accuracy of Learn From Scratch Model for 1 run: 94.93600130081177
Testing Accuracy of Learn From Scratch Model for 1 run: 86.55999898910522

Training Time Comparison

It took 3175.482 seconds to run the learn from scratch model for 125 epochs while the transfer learning model that was run three times had an execution time of a combined (1388.106+949.784+509.673) 2847.563 seconds. This is comparatively less because of less epochs. Also, learn from scratch would take more time as it is a model defined completely.

Experimental Settings: Use google colab with GPU as runtime. Create directories – vgg16 and vgg16/model to store the weights. 15 and 125 epochs are run for transfer learning and learn from scratch model respectively. Batch size is 128.

Observations

- The training accuracies for transfer learning model for all three runs were marginally different over 15 epochs. Similarly, the validation accuracy and testing accuracy were similar is variation. Losses for all three were almost similar too. This was the case with learn from scratch model too. Although the training and validation accuracies of both models were similar at 95%, testing accuracy was around 86%.
- For the pre trained model it roughly took 15 epochs to reach the desired test accuracy of 85 %. But fot the model which had to be trained from the scratch it took 125 epochs which is ro ughly 8 times more.

- The pretrained models needed less hyperparameter tunig as compared to the learn from scr atch model. Learn from scratch took lot of time and effort to tune and get to the desired acc uracy.

- Also, it took more time to fit the vgg16 from scratch than the pretrained model.

- In general both types of models were overfitting but the learn from scratch was overfitting a lot than compared with the pre trained one because the pretrained has already learned all t he features from the imagenet dataset so it doesent need to learn much or to change their weights much , wherre as other has to learn everything, every relation from scratch so it will perfectly learn only the train data due to high number of filters that this model uses.

- The other cause of overfitting could also be the small size of images (32*32*3). The vgg16 m odels has very large filter size it can easily learn the underlying relation between the input an d target variable.

- Even though the weights used by pretrained model is from different type of dataset it has be en successfully used for this problem by just changing the last fully connected layers.

- Training accuracies have been increased to over 85% for both the transfer learning model an d the learn from scratch model.

Screenshots:

Transfer Learning

First Run

```
Epoch 00013: val_accuracy improved from 0.96606 to 0.97064, saving model to /content/drive/My Drive/vgg16/checkpoint1.hdf5
391/391 [==============================] - 29s 74ms/step - loss: 0.2539 - accuracy: 0.9306 - val_loss: 0.0990 - val_accuracy: 0.9706
Epoch 14/15
391/391 [==============================] - ETA: 0s - loss: 0.2126 - accuracy: 0.9435
Epoch 00014: val_accuracy improved from 0.97064 to 0.97846, saving model to /content/drive/My Drive/vgg16/checkpoint1.hdf5
391/391 [==============================] - 29s 75ms/step - loss: 0.2126 - accuracy: 0.9435 - val_loss: 0.0809 - val_accuracy: 0.9785
Epoch 15/15
391/391 [==============================] - ETA: 0s - loss: 0.1843 - accuracy: 0.9526
Epoch 00015: val_accuracy did not improve from 0.97846
391/391 [==============================] - 29s 73ms/step - loss: 0.1843 - accuracy: 0.9526 - val_loss: 0.0977 - val_accuracy: 0.9746
```

```
[15] np.mean(history.history['accuracy']) # numpy assumed imported as np

0.7251599987347921
```

```
[16] history.history['accuracy'][14] # training accuracy of the model for the last epoch

0.9526000022888184
```

Second Run

```
Epoch 13/15
391/391 [==============================] - ETA: 0s - loss: 0.1540 - accuracy: 0.9591
Epoch 00013: val_accuracy improved from 0.97814 to 0.97898, saving model to /content/drive/My Drive/vgg16/checkpoint2.hdf5
391/391 [==============================] - 29s 75ms/step - loss: 0.1540 - accuracy: 0.9591 - val_loss: 0.0870 - val_accuracy: 0.9790
Epoch 14/15
391/391 [==============================] - ETA: 0s - loss: 0.1358 - accuracy: 0.9644
Epoch 00014: val_accuracy improved from 0.97898 to 0.98316, saving model to /content/drive/My Drive/vgg16/checkpoint2.hdf5
391/391 [==============================] - 29s 75ms/step - loss: 0.1358 - accuracy: 0.9644 - val_loss: 0.0611 - val_accuracy: 0.9832
Epoch 15/15
391/391 [==============================] - ETA: 0s - loss: 0.1236 - accuracy: 0.9676
Epoch 00015: val_accuracy improved from 0.98316 to 0.98840, saving model to /content/drive/My Drive/vgg16/checkpoint2.hdf5
391/391 [==============================] - 29s 74ms/step - loss: 0.1236 - accuracy: 0.9676 - val_loss: 0.0452 - val_accuracy: 0.9884
```

```
[24] np.mean(history2.history['accuracy'])

0.8500706712404887
```

```
[25] history2.history['accuracy'][14]

0.9676399827003479
```

Third Run

```
Epoch 13/15
391/391 [==============================] - ETA: 0s - loss: 0.1720 - accuracy: 0.9551
Epoch 00013: val_accuracy did not improve from 0.98188
391/391 [==============================] - 29s 73ms/step - loss: 0.1720 - accuracy: 0.9551 - val_loss: 0.0690 - val_accuracy: 0.9805
Epoch 14/15
391/391 [==============================] - ETA: 0s - loss: 0.1463 - accuracy: 0.9615
Epoch 00014: val_accuracy did not improve from 0.98188
391/391 [==============================] - 29s 73ms/step - loss: 0.1463 - accuracy: 0.9615 - val_loss: 0.0776 - val_accuracy: 0.9794
Epoch 15/15
391/391 [==============================] - ETA: 0s - loss: 0.1380 - accuracy: 0.9633
Epoch 00015: val_accuracy improved from 0.98188 to 0.98974, saving model to /content/drive/My Drive/vgg16/checkpoint3.hdf5
391/391 [==============================] - 29s 75ms/step - loss: 0.1380 - accuracy: 0.9633 - val_loss: 0.0374 - val_accuracy: 0.9897
```

```
[32] np.mean(history3.history['accuracy'])

     0.8586173276106517
```

```
[33] history3.history['accuracy'][14]

     0.9633399844169617
```

Average Training and Testing Accuracies for all three runs of Transfer Learning Model

1. **Transfer Learning Model**

   d. Average of Training and Testing Accuracies for 3 runs

```
[65] (((history.history['accuracy'][14])+(history2.history['accuracy'][14])+(history3.history['accuracy'][14]))/3)*100

     96.1193323135376
```

```
[37] pre_trained_model1.load_weights(os.path.join(weight_dir,'checkpoint1.hdf5'))#loading weight of a mo
     #testing model1
     score1 = pre_trained_model1.evaluate_generator(test_generator,
                              verbose=1,workers=-1,callbacks = callbacks)
     print('Test loss for first run:', score1[0])
     print('Test accuracy for first run:', score1[1])

     WARNING:tensorflow:From <ipython-input-37-d3d4e46bbe6e>:4: Model.evaluate_generator (from tensorflo
     Instructions for updating:
     Please use Model.evaluate, which supports generators.
     79/79 [==============================] - 2s 20ms/step - loss: 0.7522 - accuracy: 0.8573
     Test loss for first run: 0.7521984577178955
     Test accuracy for first run: 0.8572999835014343
```

```
[38] pre_trained_model2.load_weights(os.path.join(weight_dir,'checkpoint2.hdf5'))#loading weights of model2

     #testing model2
     score2 = pre_trained_model2.evaluate_generator(test_generator,
                              verbose=1,workers=-1,callbacks = callbacks)
     print('Test loss for second run:', score2[0])
     print('Test accuracy for second run:', score2[1])

     79/79 [==============================] - 1s 17ms/step - loss: 0.9187 - accuracy: 0.8600
     Test loss for second run: 0.9187352657318115
     Test accuracy for second run: 0.8600000143051147
```

```
[39] pre_trained_model3.load_weights(os.path.join(weight_dir,'checkpoint3.hdf5'))
     #testing model3
     score3 = pre_trained_model3.evaluate_generator(test_generator,
                              verbose=1,workers=-1,callbacks = callbacks)
     print('Test loss for third run:', score3[0])
     print('Test accuracy for third run:', score3[1])
```

```
79/79 [==============================] - 1s 17ms/step - loss: 0.7465 - accuracy: 0.8611
Test loss for third run: 0.7465143799781799
Test accuracy for third run: 0.8611000180244446
```

```
[40] #Averaging Testing scores of all the three Runs
     avg = (score1[1]+score2[1]+score3[1])/3
     print('Average test accuracies are: {:.3f} %'.format(avg*100))
```

```
Average test accuracies are: 85.947 %
```

Transfer Learning

```
Epoch 122/125
391/391 [==============================] - ETA: 0s - loss: 0.1843 - accuracy: 0.9488
Epoch 00122: val_accuracy did not improve from 0.98778
391/391 [==============================] - 33s 83ms/step - loss: 0.1843 - accuracy: 0.9488 - val_loss: 0.0468 - val_accuracy: 0.9866
Epoch 123/125
391/391 [==============================] - ETA: 0s - loss: 0.1967 - accuracy: 0.9451
Epoch 00123: val_accuracy did not improve from 0.98778
391/391 [==============================] - 33s 83ms/step - loss: 0.1967 - accuracy: 0.9451 - val_loss: 0.0426 - val_accuracy: 0.9878
Epoch 124/125
391/391 [==============================] - ETA: 0s - loss: 0.1875 - accuracy: 0.9479
Epoch 00124: val_accuracy improved from 0.98778 to 0.98940, saving model to /content/drive/My Drive/vgg16/model2/checkpoint.hdf5
391/391 [==============================] - 33s 85ms/step - loss: 0.1875 - accuracy: 0.9479 - val_loss: 0.0393 - val_accuracy: 0.9894
Epoch 125/125
391/391 [==============================] - ETA: 0s - loss: 0.1843 - accuracy: 0.9494
Epoch 00125: val_accuracy did not improve from 0.98940
391/391 [==============================] - 32s 83ms/step - loss: 0.1843 - accuracy: 0.9494 - val_loss: 0.0607 - val_accuracy: 0.9823
```

```
[47] #loading model weights
     model2.load_weights(os.path.join(weight_dir,'model2/checkpoint.hdf5'))
```

```
[48] #testing on test data
     score = model2.evaluate_generator(test_generator,
                              verbose=1,workers=-1,callbacks = callbacks2)
     print('Test loss:', score[0])
     print('Test accuracy:', score[1])
```

```
79/79 [==============================] - 2s 20ms/step - loss: 0.5767 - accuracy: 0.8656
Test loss: 0.5767048001289368
Test accuracy: 0.8655999898910522
```

```
[49] hist_scratch.history['accuracy'][124] # 125th epoch accuracy
```

```
0.9493600130081177
```

```
[50] np.mean(hist_scratch.history['accuracy']) # numpy assumed imported as np
```

```
0.8004679992198944
```

TASK-3: COMPARING MODELS:

Comparison of Accuracies for Transfer Learning and Learn from Scratch models

```
[62] print("Transfer Learning Model")
     print("Average Training Accuracy of Transfer Learning Model for 3 runs:",(((history.history['accuracy'][14])+(history2.history['accuracy'][14])+(history3.history['accur
     print("Average Testing Accuracy of Transfer Learning Model for 3 runs:",((score1[1]+score2[1]+score3[1])/3)*100)
     print("Learn From Scratch Model")
     print("Training Accuracy of Learn From Scratch Model for 1 run:",(hist_scratch.history['accuracy'][124]*100))
     print("Testing Accuracy of Learn From Scratch Model for 1 run:",(score[1]*100))
```

```
Transfer Learning Model
Average Training Accuracy of Transfer Learning Model for 3 runs: 96.1193323135376
Average Testing Accuracy of Transfer Learning Model for 3 runs: 85.94666719436646
Learn From Scratch Model
Training Accuracy of Learn From Scratch Model for 1 run: 94.93600130081177
Testing Accuracy of Learn From Scratch Model for 1 run: 86.55999898910522
```

Execution times

```
callbacks = [checkpoint,reduce_lr,earlystop]
```

```
#fitting model
1.fit(train_generator,validation_data=valid_generator,verbose=1,workers=-1,callbacks=callbacks,epochs = 15)
```

Run cell (Ctrl+Enter)
cell executed since last change

executed by Khushal Paresh Thaker
17:36 (2 hours ago)
executed in 509.673 s

```
============] - ETA: 0s - loss: 2.2224 - accuracy: 0.1544
mproved from -inf to 0.25976, saving model to /content/drive/My Drive/vgg16/checkpoint1.hdf5
============] - 28s 73ms/step - loss: 2.2224 - accuracy: 0.1544 - val_loss: 1.9326 - val_accuracy: 0.2598

============] - ETA: 0s - loss: 1.9760 - accuracy: 0.2621
Epoch 00002: val_accuracy improved from 0.25976 to 0.55292, saving model to /content/drive/My Drive/vgg16/checkpoint1.hdf5
391/391 [==============================] - 29s 73ms/step - loss: 1.9760 - accuracy: 0.2621 - val_loss: 1.5186 - val_accuracy: 0.5529
Epoch 3/15
391/391 [==============================] - ETA: 0s - loss: 1.5966 - accuracy: 0.4242
```

```
#fitting transfer learning model for second run
12.fit(train_generator,validation_data=valid_generator,verbose=1,workers=-1,callbacks=callbacks,epochs = 15)
```

Run cell (Ctrl+Enter)
cell executed since last change

executed by Khushal Paresh Thaker
17:43 (2 hours ago)
executed in 949.784 s

```
============] - ETA: 0s - loss: 1.8232 - accuracy: 0.3479
mproved from -inf to 0.70594, saving model to /content/drive/My Drive/vgg16/checkpoint2.hdf5
============] - 29s 74ms/step - loss: 1.8232 - accuracy: 0.3479 - val_loss: 0.9881 - val_accuracy: 0.7059

============] - ETA: 0s - loss: 1.0809 - accuracy: 0.6643
Epoch 00002: val_accuracy improved from 0.70594 to 0.82236, saving model to /content/drive/My Drive/vgg16/checkpoint2.hdf5
391/391 [==============================] - 29s 74ms/step - loss: 1.0809 - accuracy: 0.6643 - val_loss: 0.5863 - val_accuracy: 0.8224
Epoch 3/15
391/391 [==============================] - ETA: 0s - loss: 0.8159 - accuracy: 0.7593
Epoch 00003: val_accuracy improved from 0.82236 to 0.85720, saving model to /content/drive/My Drive/vgg16/checkpoint2.hdf5
391/391 [==============================] - 29s 75ms/step - loss: 0.8159 - accuracy: 0.7593 - val_loss: 0.4664 - val_accuracy: 0.8572
Epoch 4/15
```

```
#fitting 3rd model
13.fit(train_generator,validation_data=valid_generator,verbose=1,workers=-1,callbacks=callbacks,epochs = 15
    )
```

Run cell (Ctrl+Enter)
cell executed since last change

executed by Khushal Paresh Thaker
17:51 (2 hours ago)
executed in 1388.106 s

```
mproved from -inf to 0.73102, saving model to /content/drive/My Drive/vgg16/checkpoint3.hdf5
============] - 29s 74ms/step - loss: 1.6330 - accuracy: 0.4451 - val_loss: 0.9110 - val_accuracy: 0.7310

============] - ETA: 0s - loss: 1.0246 - accuracy: 0.6979
Epoch 00002: val_accuracy improved from 0.73102 to 0.82458, saving model to /content/drive/My Drive/vgg16/checkpoint3.hdf5
391/391 [==============================] - 29s 74ms/step - loss: 1.0246 - accuracy: 0.6979 - val_loss: 0.6216 - val_accuracy: 0.8246
Epoch 3/15
391/391 [==============================] - ETA: 0s - loss: 0.7978 - accuracy: 0.7691
Epoch 00003: val_accuracy improved from 0.82458 to 0.85970, saving model to /content/drive/My Drive/vgg16/checkpoint3.hdf5
391/391 [==============================] - 29s 74ms/step - loss: 0.7978 - accuracy: 0.7691 - val_loss: 0.4801 - val_accuracy: 0.8597
```

```
#trinaing the model from scratch
rain_generator,validation_data=valid_generator,verbose=1,workers=-1,callbacks=callbacks2,epochs = 125)
```

Run cell (Ctrl+Enter)
cell executed since last change

executed by Khushal Paresh Thaker
18:58 (1 hour ago)
executed in 3175.482 s

```
============] - ETA: 0s - loss: 2.6124 - accuracy: 0.1366
mproved from -inf to 0.10004, saving model to /content/drive/My Drive/vgg16/model2/checkpoint.hdf5
============] - 32s 81ms/step - loss: 2.6124 - accuracy: 0.1366 - val_loss: 3.1356 - val_accuracy: 0.1000
391/391 [==============================] - ETA: 0s - loss: 2.3109 - accuracy: 0.1772
Epoch 00002: val_accuracy did not improve from 0.10004
391/391 [==============================] - 31s 79ms/step - loss: 2.3109 - accuracy: 0.1772 - val_loss: 5.5837 - val_accuracy: 0.0999
Epoch 3/125
391/391 [==============================] - ETA: 0s - loss: 2.1990 - accuracy: 0.1968
Epoch 00003: val_accuracy did not improve from 0.10004
391/391 [==============================] - 30s 78ms/step - loss: 2.1990 - accuracy: 0.1968 - val_loss: 3.2604 - val_accuracy: 0.0919
Epoch 4/125
391/391 [==============================] - ETA: 0s - loss: 2.1179 - accuracy: 0.2127
Epoch 00004: val_accuracy did not improve from 0.10004
```