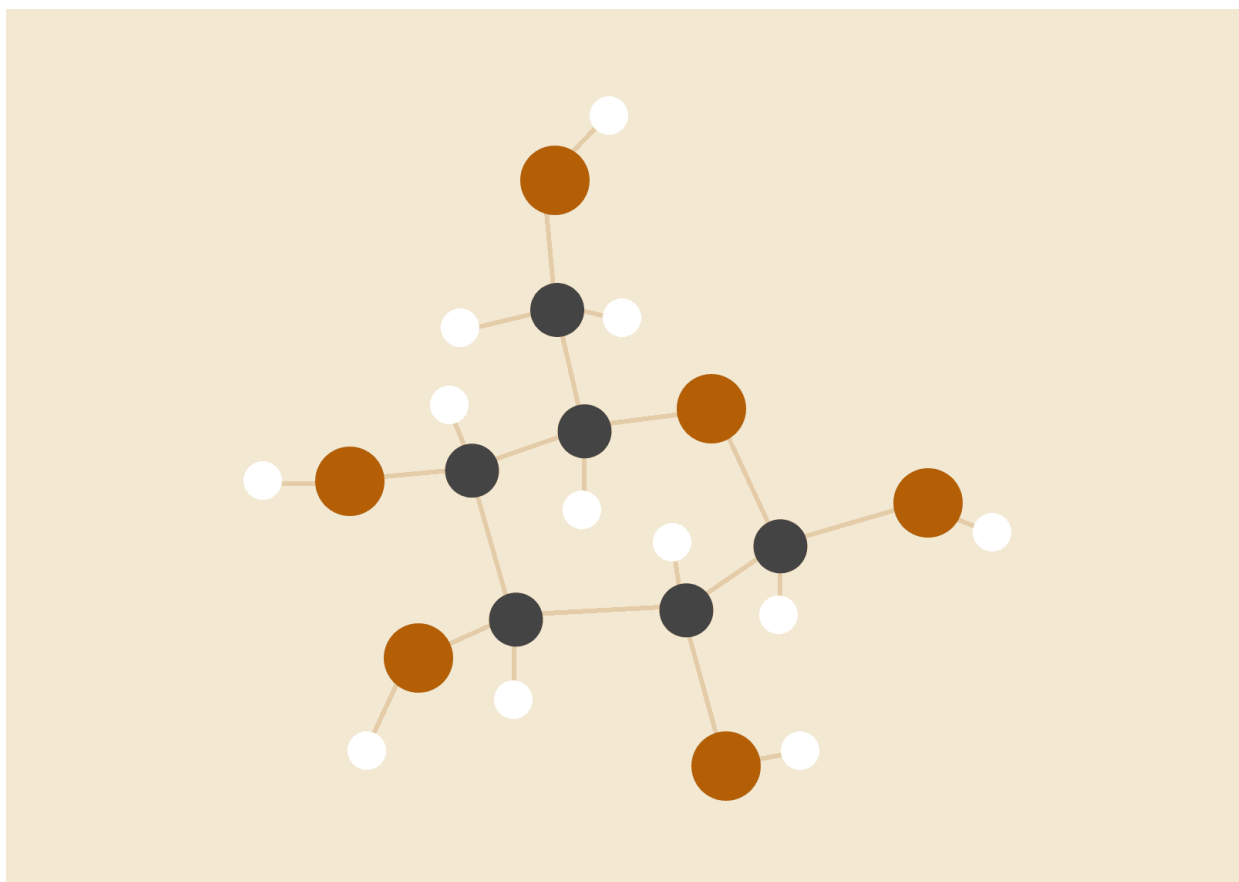# Assignment-1(CSE676)

Deep Learning Assignment

Suryaveer Singh

04/07/2023

INTRODUCTION

I have created a deep learning model trying to replicate the efficient architecture of MobileNet. I have successfully trained the model to have training accuracy =59% and testing accuracy = 58%

I have also modified the model to showcase overfitting and underfitting successfully.

Task 1

Desc: Successfully train a Convolutional Neural Network (CNN) to classify an image dataset with textures found in the wild.

```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1        [-1, 32, 112, 112]             864
       BatchNorm2d-2        [-1, 32, 112, 112]              64
              ReLU-3        [-1, 32, 112, 112]               0
            Conv2d-4        [-1, 32, 112, 112]             288
       BatchNorm2d-5        [-1, 32, 112, 112]              64
              ReLU-6        [-1, 32, 112, 112]               0
            Conv2d-7        [-1, 64, 112, 112]           2,048
       BatchNorm2d-8        [-1, 64, 112, 112]             128
              ReLU-9        [-1, 64, 112, 112]               0
           Conv2d-10          [-1, 64, 56, 56]             576
      BatchNorm2d-11          [-1, 64, 56, 56]             128
             ReLU-12          [-1, 64, 56, 56]               0
           Conv2d-13         [-1, 128, 56, 56]           8,192
      BatchNorm2d-14         [-1, 128, 56, 56]             256
             ReLU-15         [-1, 128, 56, 56]               0
           Conv2d-16         [-1, 128, 56, 56]           1,152
      BatchNorm2d-17         [-1, 128, 56, 56]             256
             ReLU-18         [-1, 128, 56, 56]               0
           Conv2d-19         [-1, 128, 56, 56]          16,384
      BatchNorm2d-20         [-1, 128, 56, 56]             256
             ReLU-21         [-1, 128, 56, 56]               0
           Conv2d-37         [-1, 512, 14, 14]         131,072
      BatchNorm2d-38         [-1, 512, 14, 14]           1,024
             ReLU-39         [-1, 512, 14, 14]               0
AdaptiveAvgPool2d-40           [-1, 512, 1, 1]               0
          Dropout-41           [-1, 512, 1, 1]               0
           Linear-42                  [-1, 47]          24,111
================================================================
Total params: 293,231
Trainable params: 293,231
Non-trainable params: 0
```

Model details :

Loss function chosen :

I chose a cross entropy loss function to calculate loss as it is the preferred loss function for classification tasks using CNN's. I also experimented with Negative log likelihood loss function but was unable to get fast learning in initial epochs.

Optimizer chosen for backward propagation:

I chose Adam optimizer for back propagation with learning rate 0.01. Changing the learning rate to 0.01 and 0.001 was not at all helpful as the most optimum lr for adam is 0.01.
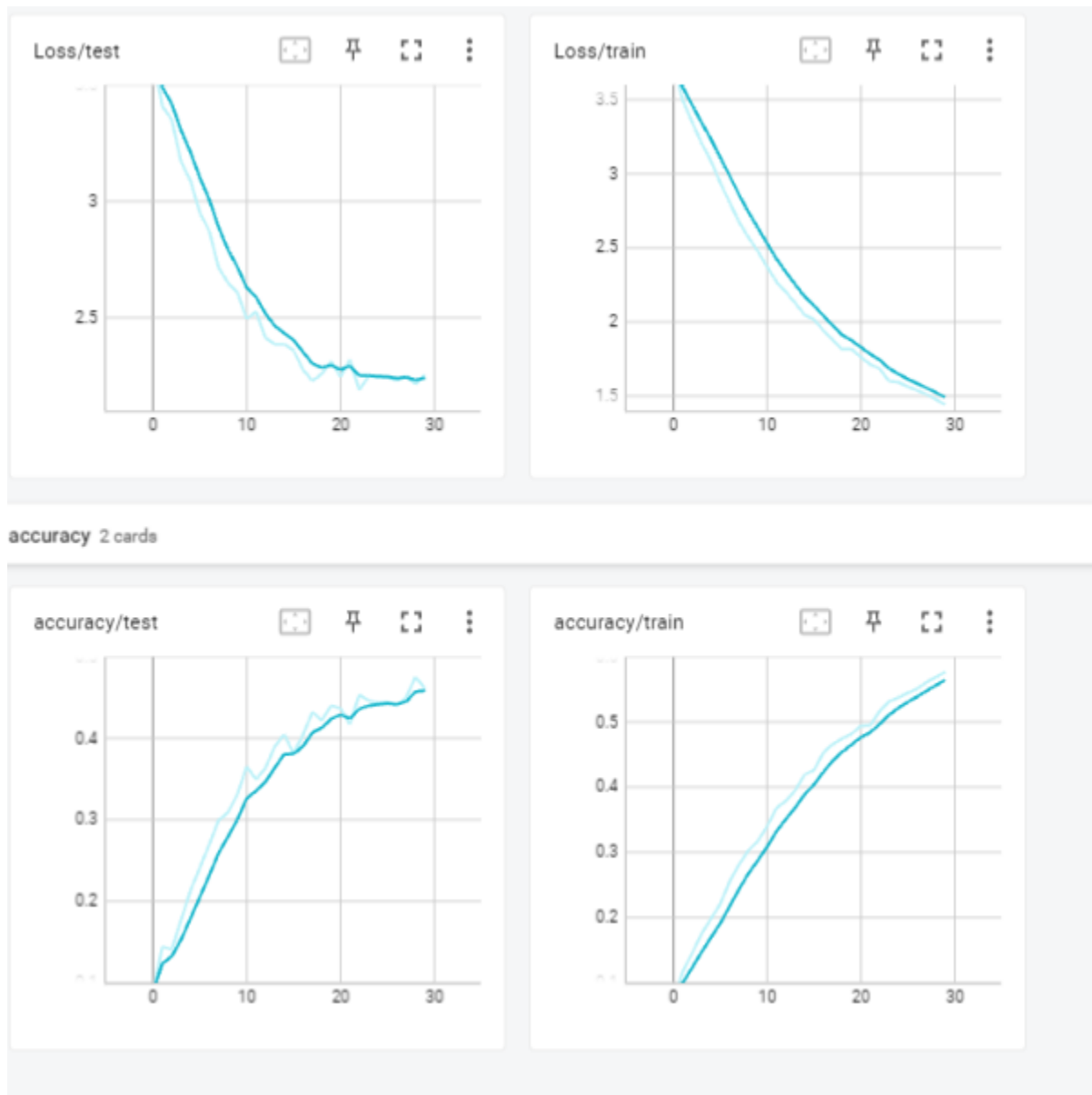
BatchSize:

Batch size for training and testing data used was 16.

Accuracy:

Training: 59%

Testing: 58%

Generalization error against number of epochs:

Visualizing weights:

Task 2 : UnderFitting

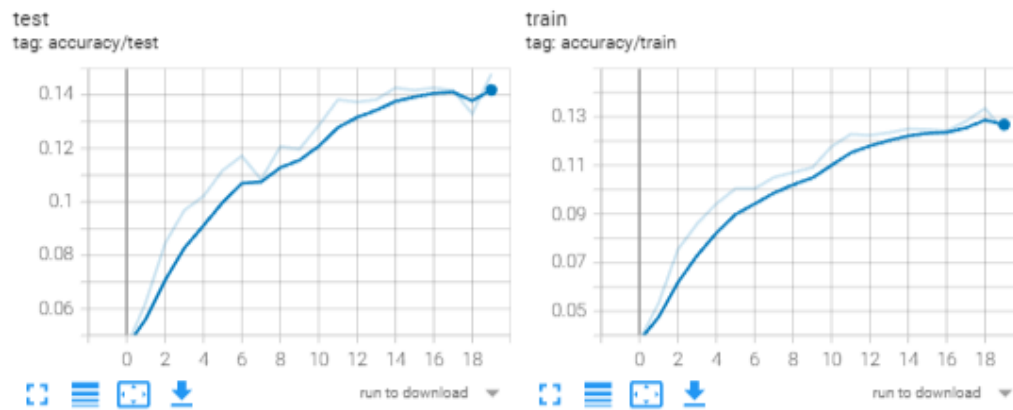To do underfitting I reduced the number of layers to and reduced epochs to 20.

The model stops at nearest minima and does not train further to find global minima of loss and is under- fitted with training accuracy of 12% and validation accuracy of 14%.
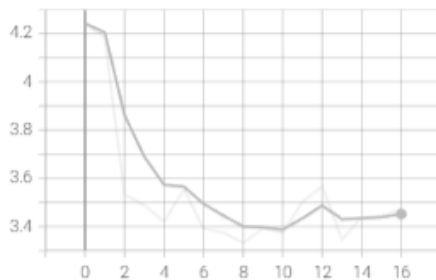
Graphs:

Loss

test
tag: Loss/test

train
tag: Loss/train

accuracy

test
tag: accuracy/test

train
tag: accuracy/train

Task3 : Overfitting

Added extra epochs to make the model overfit. After 10 epochs the loss graph turns positive and loss starts increasing. Training accuracy we got was 35% and testing accuracy was 22%.
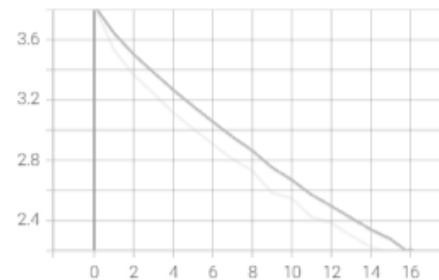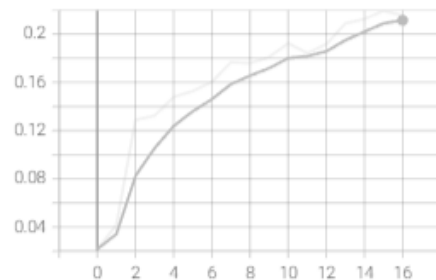
Graphs: