Deep Learning Project - 2

Kishalay Das (SR Number-15938)

February 21, 2020

Classification using CNN and MLP :

Task :

This project is to implement neural network and convolutional neural network for the task of classifi- cation. The classification task will be that of recognizing an image and identify it as one of ten classes. You are required to train the classifiers using Fashion-MNIST clothing images. Following are the two tasks to be performed:

1. Build multi-layer Neural Network with open-source neural-network library on Fashion-MNIST dataset.
2. Build Convolutional Neural Network with open-source neural-network library on Fashion-MNIST dataset.

Approach 1 : Train using Multi-Layer Neural Network

I have build a Multi-Layer Neural Network using pytorch library function with input dimension (28*28=784),output dimension 10 and three hidden layers with dimensions 128,64,32 respectively.Then we optimise the parameters of the model using Adam optimiser with learning rate 0.001 and train it for 25 epocs.We have kept the batch size as 4 and activation function as relu.
I split the train set into 80-20 percent train-validation set and check the validation accuracy.Also I have kept early stopping with patience coefficient as 10 epocs to avoid overfitting. We tried different configurations as a part of experiments and we observed following train-loss and validation accuracy results:

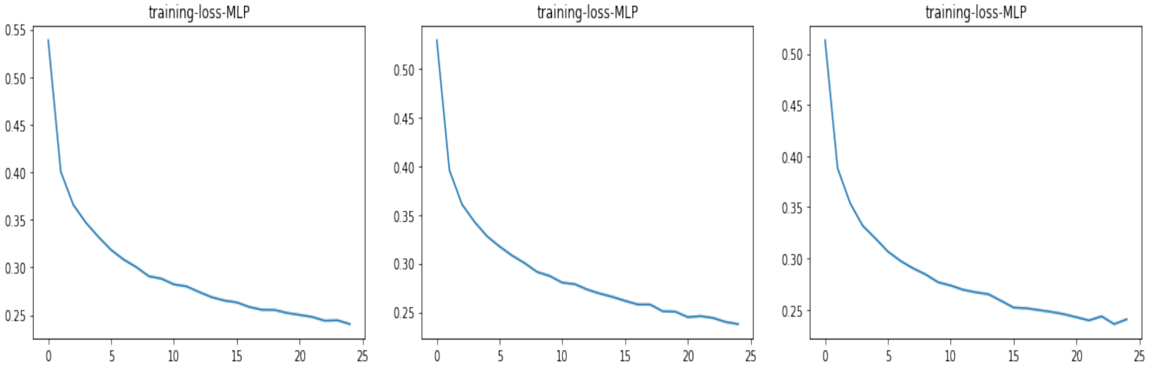


Fig1: Train Loss plot for MLP Hidden configuration 128/64/32, 64/32/16, 256/64/32

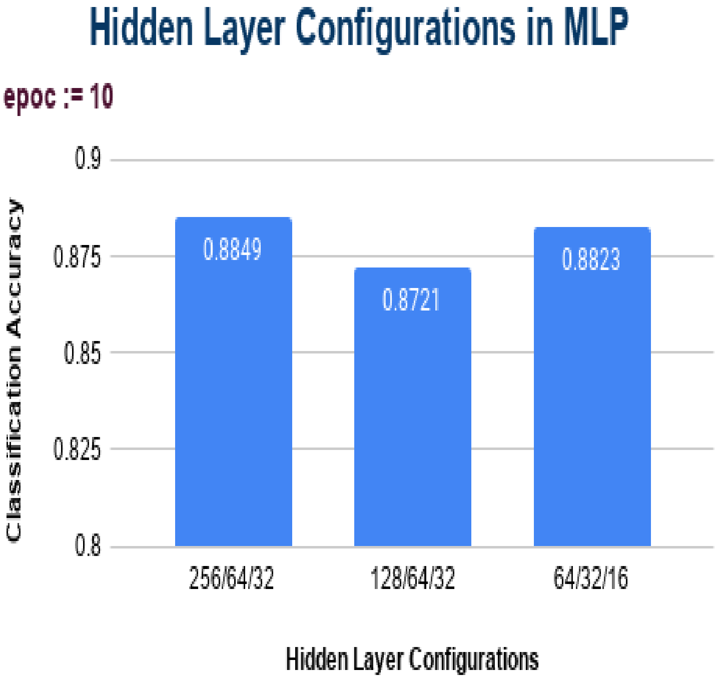
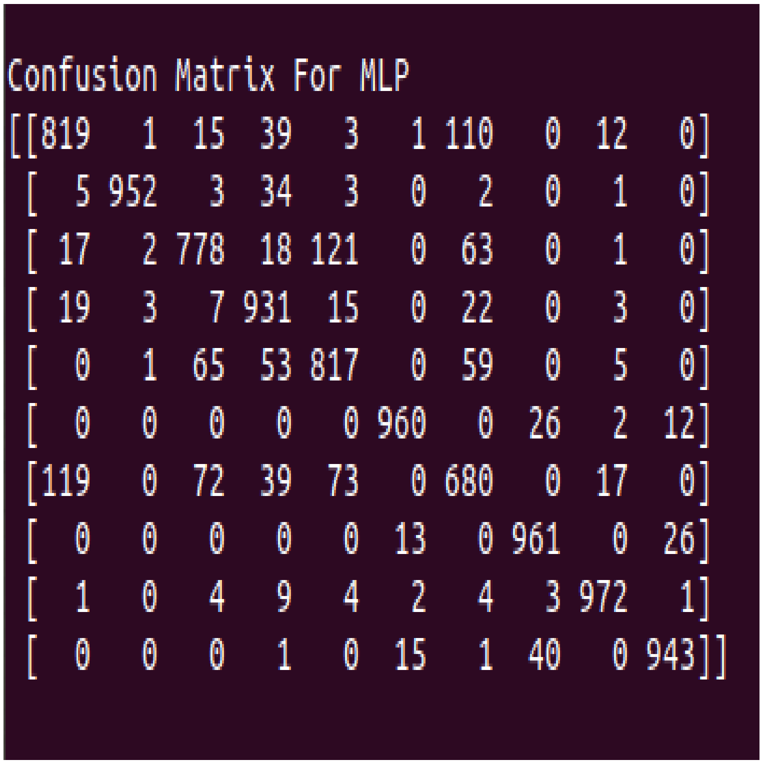


Fig1: Validation Accuracy of model for different configurations for MLP.

As per the plot given above for different architecture configurations I have chosen 128,64,32 as final configurations because it is performing better on validation sets.

Relative strengths and weaknesses of the model:

Following is the confusion matrix of classifier on test data:



```
Confusion Matrix For MLP
[[819   1  15  39   3   1 110   0  12   0]
 [  5 952   3  34   3   0   2   0   1   0]
 [ 17   2 778  18 121   0  63   0   1   0]
 [ 19   3   7 931  15   0  22   0   3   0]
 [  0   1  65  53 817   0  59   0   5   0]
 [  0   0   0   0   0 960   0  26   2  12]
 [119   0  72  39  73   0 680   0  17   0]
 [  0   0   0   0   0  13   0 961   0  26]
 [  1   0   4   9   4   2   4   3 972   1]
 [  0   0   0   1   0  15   1  40   0 943]]
```

There are few observations found in the confusion matrix.Like from first row,True positive count is 819 but with class 7 false positive count is 110.And with other category the miss-classification count is pretty low.The reason behind it is class 1 is T-shirt/Top and class 7 is only shirt.So as they are close enough so we are getting few miss-classification.Similar observation found for Class-3(Pullover) and Class-5(Coat).Otherwise the model is strong enough to classify data items from different classes.

Approach 2 : Train using Convolution Neural Network

We build a Convolution Neural Network using pytorch library function with two convolution layers with dimension 32 and 64 respectively.After each convolution layer I have performed a Batch Normalisation, then relu non-linearity and finally max pooling.Then we flatten the output and pass it through two fully connected layers of dimension 512(Embedding Dimension) and 128 respectively.Again as mentioned abovewe optimise the parameters of the model using Adam optimiser with learning rate 0.001 and train it for 25 epocs.We have kept the batch size as 4 and activation function as relu.

I split the train set into 80-20 percent train-validation set and check the validation accuracy.Also I have kept early stopping with patience coefficient as 10 epocs to avoid overfitting. We tried different configurations as a part of experiments and we observed following train-loss and validation accuracy results:

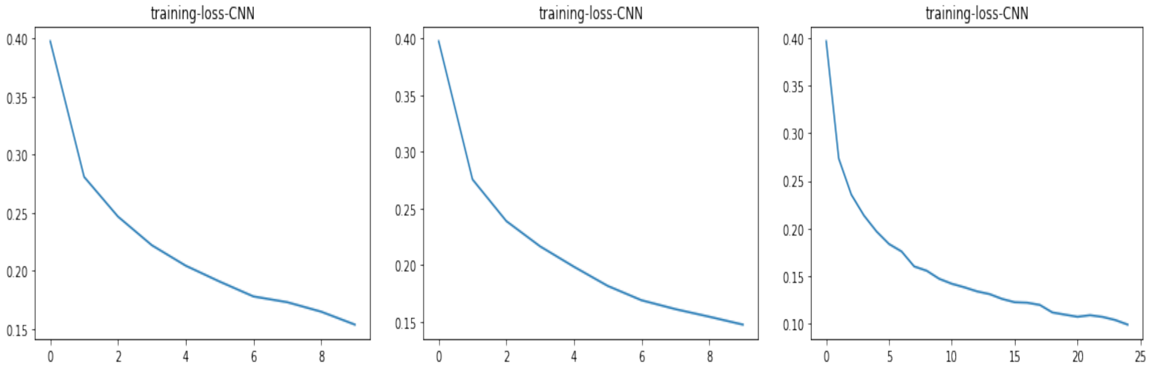


Fig1: Train Loss plot for CNN Hidden configuration 16/32, 32/64, 64/128

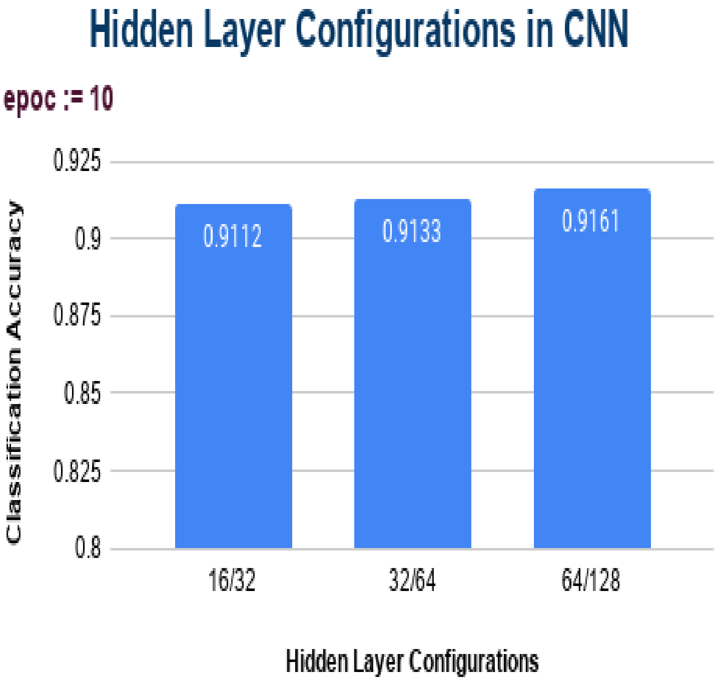
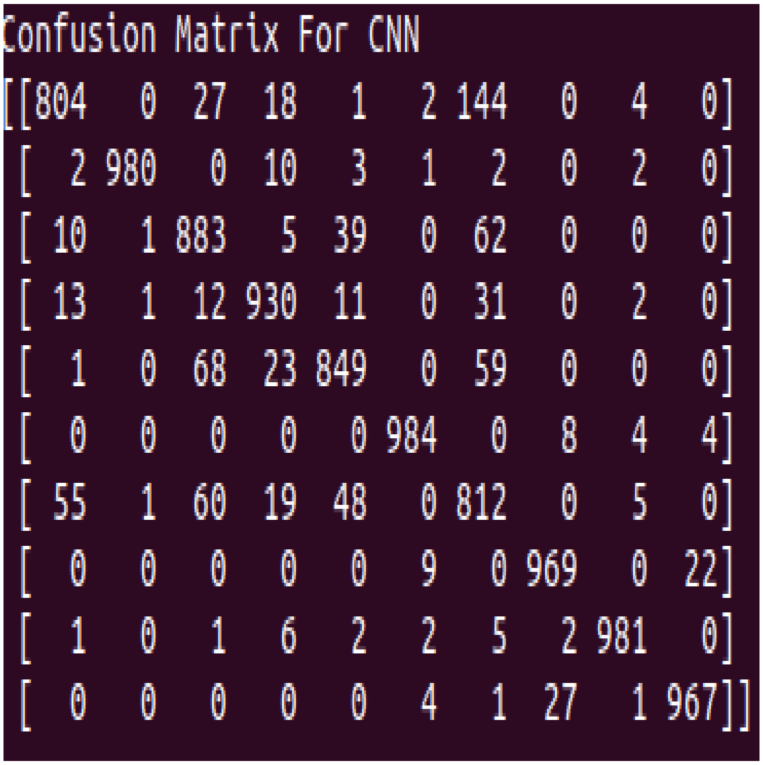


Fig1: Validation Accuracy of model for different configurations for CNN.

As per the plot given above for different architecture configurations I have chosen 32,64 as final configurations because it is performing better on validation sets. Following is the confusion matrix of classifier on test data:



```
Confusion Matrix For CNN
[[804   0  27  18   1   2 144   0   4   0]
 [  2 980   0  10   3   1   2   0   2   0]
 [ 10   1 883   5  39   0  62   0   0   0]
 [ 13   1  12 930  11   0  31   0   2   0]
 [  1   0  68  23 849   0  59   0   0   0]
 [  0   0   0   0   0 984   0   8   4   4]
 [ 55   1  60  19  48   0 812   0   5   0]
 [  0   0   0   0   0   9   0 969   0  22]
 [  1   0   1   6   2   2   5   2 981   0]
 [  0   0   0   0   0   4   1  27   1 967]]
```

Relative strengths and weaknesses of the model:

There are few observations found in the confusion matrix.Overall number correct classification has improved along the diagonal of the matrix compared to simple mlp.In first row,True positive count is 804 but with class 7 false positive count is 144.And with other category the miss-classification count is pretty low.Again the reason behind it is class 1 is T-shirt/Top and class 7 is only shirt.So as they are close enough so we are getting few miss-classification.Similar observation found for Class-3(Pullover) and Class-5(Coat) though count has reduced for miss classification.Otherwise the model is strong enough to classify data items from different classes.