COMPUTER VISION AND PATTERN RECOGNITION

MIDTERM PROJECT

Name: Shadman Shakib Aditto

ID: 18-38937-3

Section: A

Answer to the question no: 03

Report based on Question no 2.

Abstract:

In this report I am going to discuss about implementing a CNN architecture to classify the MNIST handwritten dataset. Also I used different optimizer like Adam, SGD, RSMProp to check which one gives best accuracy. In this report I am going to discuss about implementing a CNN architecture to classify the MNIST handwritten dataset. Also I used different optimizer like Adam, SGD, RSMProp to check which one gives best accuracy.

Introduction:

Optimizers are techniques or approaches that adjust the characteristics of your neural network, such as weights and learning rate, to decrease losses. Optimization algorithms or methods are in charge of lowering losses and delivering the most accurate outcomes. Optimizers are algorithms or techniques for changing the properties of your neural network, such as weights and learning rate, in order to decrease losses.... Optimization algorithms or strategies are in charge of decreasing losses and providing the most accurate results feasible.

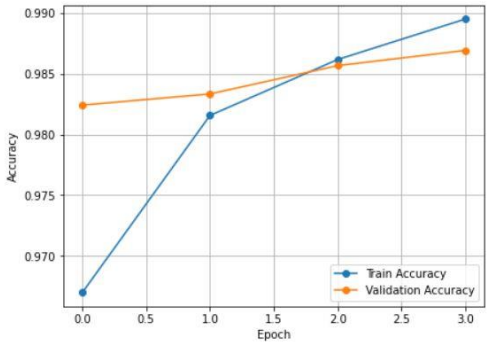
As earlier I said I will use 3 type of optimizer and now I will discuss about them:

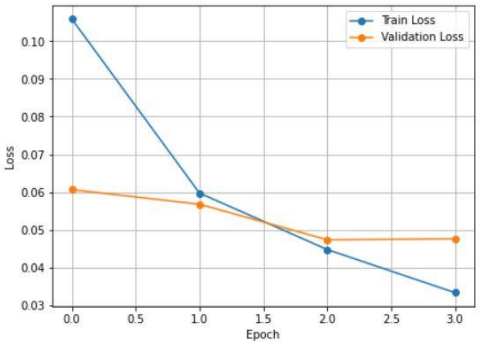
Adam: Adam is a deep learning model training technique that replaces stochastic gradient descent. Adam combines the finest features of the AdaGrad and RMSProp methods to provide an optimization technique for noisy issues with sparse gradients.

SGD: SGD is an iterative approach for finding the best smoothness qualities for an objective function. One popular and persuasive argument for optimizers is that SGD generalizes better than Adam.

RMSProp: Root Mean Square Propagation is abbreviated as RMSprop. In neural network training, RMSprop is a gradient-based optimization strategy.

Results





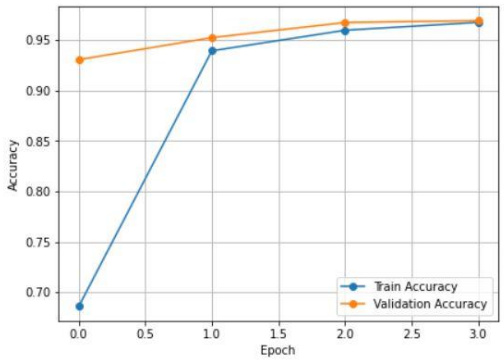

```
test_loss, test_acc = model.evaluate(X_test, Y_test)
print('\nTest Accuracy:', test_acc)
print('\nTest Loss:', test_loss)
```

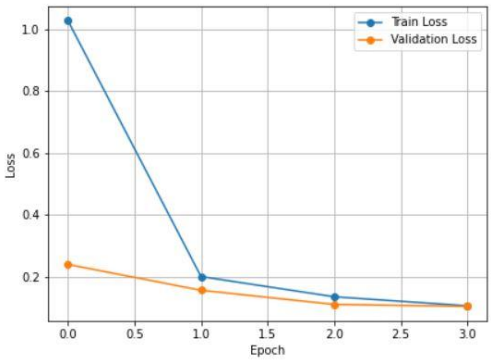
```
313/313 [==============================] - 3s 9ms/step - loss: 0.0433 - accuracy: 0.9879

Test Accuracy: 0.9879000186920166

Test Loss: 0.04332428798079491
```

Here, I use ADAM optimizer and the test accuracy is 0.9879000186920166 that means 98.79%. The test loss is 0.04332428798879491 that means 0.4%.





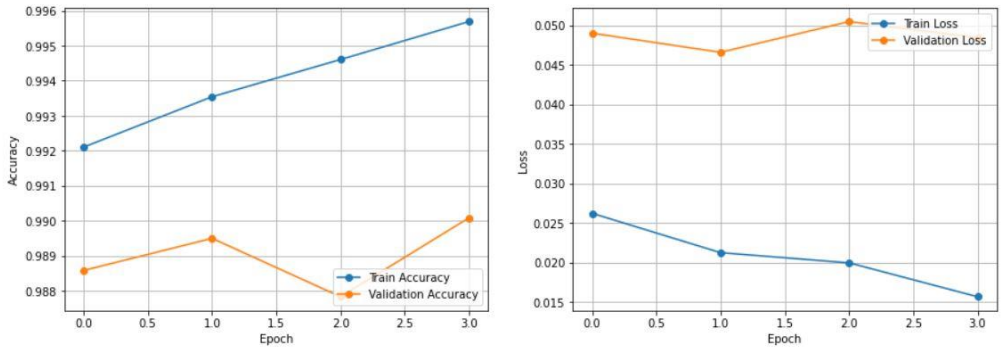

```
test_loss, test_acc = model.evaluate(X_test, Y_test)
print('\nTest Accuracy:', test_acc)
print('\nTest Loss:', test_loss)
```

```
313/313 [==============================] - 3s 9ms/step - loss: 0.0979 - accuracy: 0.9707

Test Accuracy: 0.9707000255584717

Test Loss: 0.09790098667144775
```

Here, I use SGD optimizer and the test accuracy is 0.9707000255584717 that means 97.07%. The test loss is 0.09790098667144775 that means 0.9%.




```
test_loss, test_acc = model.evaluate(X_test, Y_test)
print('\nTest Accuracy:', test_acc)
print('\nTest Loss:', test_loss)
```

```
313/313 [==============================] - 3s 9ms/step - loss: 0.0431 - accuracy: 0.9913

Test Accuracy: 0.9912999868392944

Test Loss: 0.043141648173332214
```

Here I use RMSprop optimizer and the test accuracy is 0.9912999868392944 that means 99.12%. The test loss is 0.043141648173332214 that means 0.4%.

Discussion:

This report uses three types of optimizers: ADAM, SGD, and RMS Drop. So I found a slight difference of between those precisions. ADAM is much faster than SGD and RMS Drop. Built on the strengths of previous models, the Adam Optimizer offers much higher performance than the previously used and far outperforms it by providing an optimized descent gradient. My ADAM accuracy is 98.79%. The second collaborative optimizer is the SGD, which is also good and my SGD accuracy is 97.07%. The last one is RMSProp, my RMSProp is 99.12%, which is more accurate from ADAM and SGD.