American International University- Bangladesh

Mid Project Report Fall 2021-22

NAME	KAZI SHAKAWAT ABSAR
ID	18-36913-1
COURSE	Computer Vision and Pattern Recognition
SECTION	A
DATE	30/10/2021

Abstract:

CNN is a kind of synthetic neural network used to analyze visual facts in deep gaining knowledge of. Convolutional Neural Networks are networks containing one or greater convolutional layers to investigate snap shots, classify facts, and phase it. Here I applied CNN structure to categorize the MNIST handwritten dataset inside the research. To take a look at different levels of accuracy, I carried out 3 sorts of optimizers: ADAM, SGD and RMSProp.

Introduction:

The convolutional neural community is a kind of artificial neural network that is used in photograph processing and recognition. In CNN optimizers are the algorithms or methods used to change the attributes of the neural community together with weights and studying rate for you to reduce the losses. In this venture, three kinds of optimizers had been used; ADAM, SGD and RMSProp.

ADAM is an optimization set of rules that can be used in place of the classical stochastic gradient descent process to replace network weights iterative based totally in training data. ADAM is a famous algorithm within the area of deep mastering due to the fact it may reap excellent outcomes.

SGD is an iterative technique for optimizing an objective characteristic with suitable smoothness residences. But ADAM is plenty faster than SGD.

RMSProp is a gradient-primarily based optimization technique utilized in training neural networks. This normalization balances the step length(momentum), reducing the step for large gradients to avoid exploding and increasing the step for small gradients to keep away from vanishing

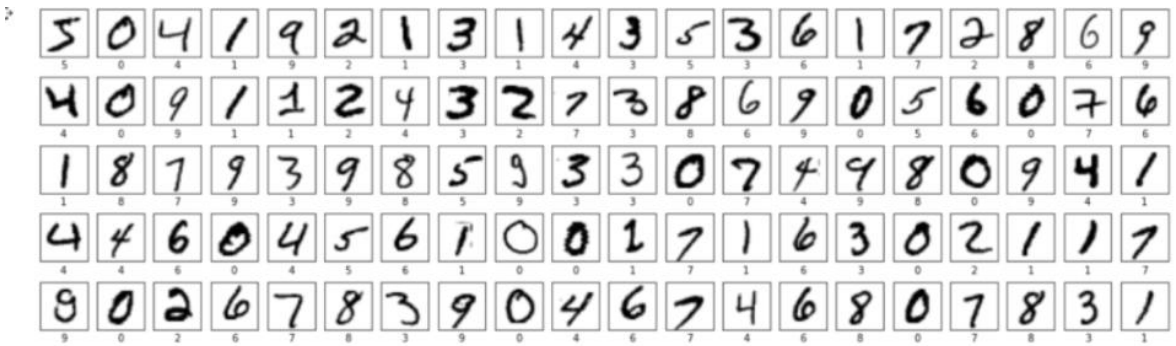
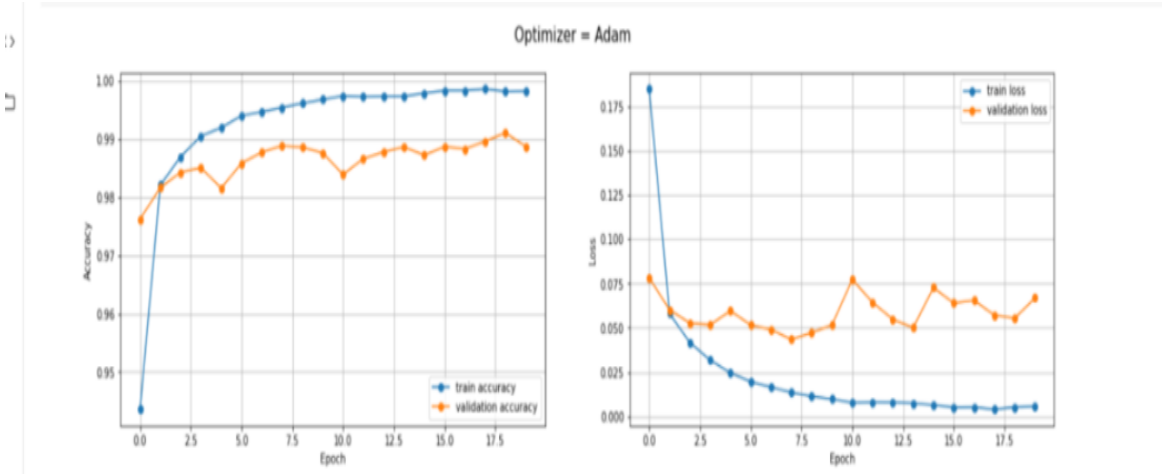


Figure 1: MNIST handwritten dataset

Result

For ADAM I achieved accuracy 99% and loss 4.91%



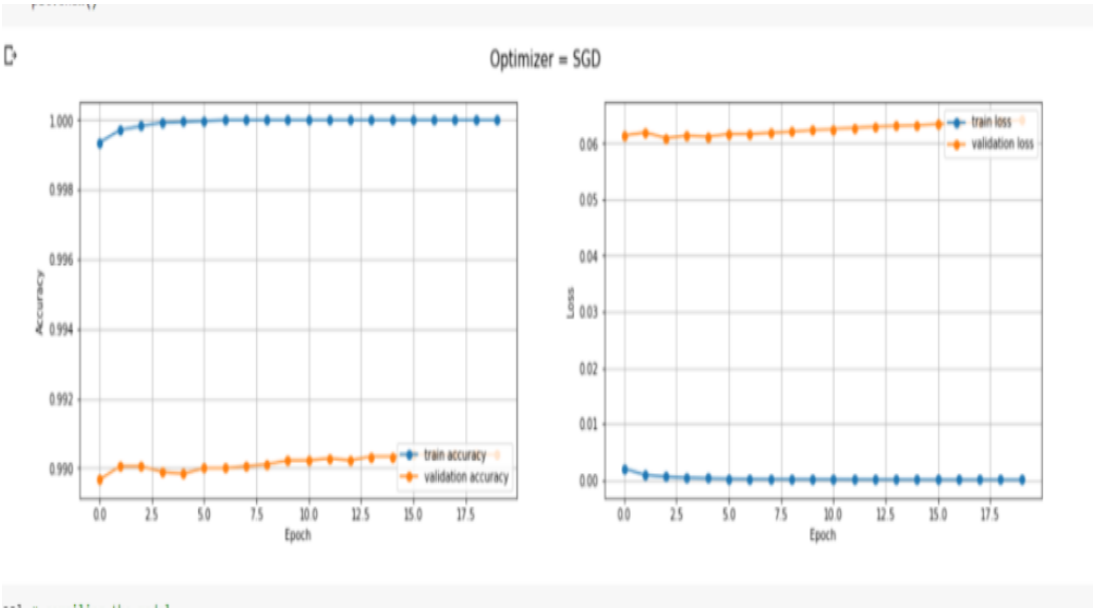
```
#test accuracy
test_loss, test_accuracy = model.evaluate(X_test, Y_test)
print(f'\nTest accuracy: {test_accuracy}')
```

```
313/313 [==============================] - 2s 6ms/step - loss: 0.0491 - accuracy: 0.9900

Test accuracy: 0.9900000095367432
```

Figure 2: Test accuracy and loss for ADAM

For SGD, I achieved accuracy 99.23% and Loss 4.38%.



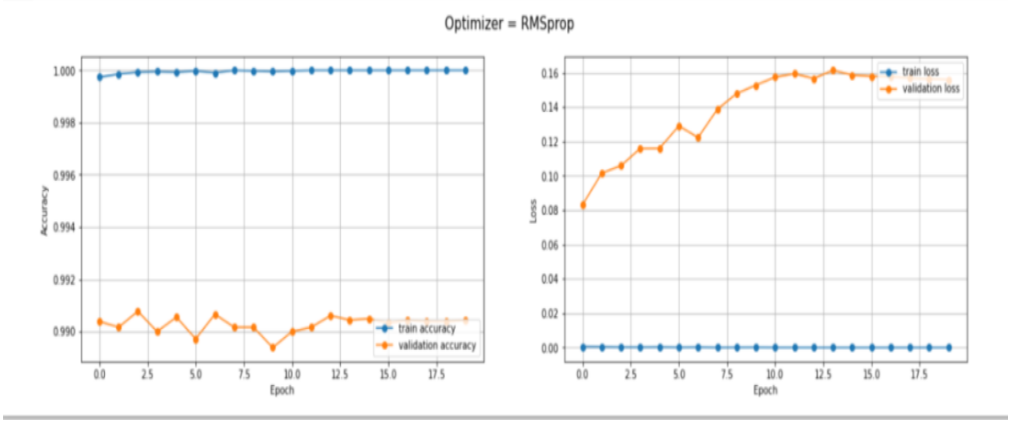

```
# training the model
test_loss, test_accuracy = model.evaluate(X_test, Y_test)
print(f'\nTest accuracy: {test_accuracy}')
```

```
313/313 [==============================] - 2s 8ms/step - loss: 0.0438 - accuracy: 0.9923

Test accuracy: 0.9922999739646912
```

Figure 3: Test accuracy and loss for SGD

For RMSProp, I achieved accuracy 99.22% and loss 10.04%




```
# training the model
test_loss, test_accuracy = model.evaluate(X_test, Y_test)
print(f'\nTest accuracy: {test_accuracy}')
```

```
313/313 [==============================] - 2s 7ms/step - loss: 0.1004 - accuracy: 0.9922

Test accuracy: 0.9922000169754028
```

Figure 4: Test accuracy and loss for RMSProp

Discussion:

Here the application of ADAM, SGD, RMSProp as optimizers were used on this task. I had experienced little dissimilarities of their precision. Compared to ADAM and RMSProp, SGD is quicker and extra powerful. The SGD optimizer outperforms the prior optimizer with the aid of a sizeable margin. The accuracy of my SGD optimizer is 99.23% and the loss is 4.38%. Then, there is ADAM which is also an amazing optimizer with a 99 percentage accuracy and a 4.91% loss. The remaining one is RMSProp, which has a 99.22% accuracy and a loss of 10.04%, that is extensively less than SGD however slightly extra than ADAM. So, in my instance, the SGD optimizer is the quickest and most unique optimizer.