Deep Learning

MULTIMODAL CLASSIFICATION ASSIGNMENT 2

FC Neural Network

1. Hyperparameters

a. Learning rate (0.01, 0.001)

b. Optimizer (sgd, adagrad, adam, rmsprop)

c. Loss

   i. Only the sparse categorical cross entropy works as the output is digit and not 1 hot encoding, else we should have used categorical cross entropy.

d. Activations functions (Sigmoid, Relu, Tanh)

   i. The activation function for the output layer must be softmax

2. (Using default LR and relu activation function) [Optimizer HP]

Optimizer	SGD	Adagrad	RMSprop	Adam
Train-Acc	0.91	0.91	0.981	0.983
Test-Acc	0.92	0.91	0.975	0.976
Graph	sgd_opt	adagrad_opt	rmsprop_opt	adam_opt

3. (Using relu activation function) [Learning rate HP]

Optimizer	RMSprop		Adam	
LR	0.01	0.01	0.01	0.001
Train-Acc	0.978	0.981	0.975	0.983
Test-Acc	0.967	0.975	0.963	0.974
Graph	rmsprop_0.01	rmsprop_0.001	adam_0.01	adam_0.001

4. (Using Adam and LR = 0.001) [Activation HP]

Activation	Sigmoid	Relu	tanh
Train-Acc	0.966	0.982	0.981
Test-Acc	0.964	0.976	0.973
Graph	sigmoid_act	relu_act	tanh_act

5. Analysis

a. Best optimizer: Adam and RMSprop with Adam slightly better

b. Best Learning rate: 0.001 with both Adam and RMSprop

c. Best Activation Function considering the best optimizer with best LR: Relu slightly better than tanh.

6. Observation

a. Test accuracy never reached the training accuracy

b. Overfitting happens some of the time

Convolution Neural Network

1. Hyperparameters

a. Activation (Relu, Tanh)

b. Filter size (3, 5, 7)

c. Stride length (1, 2, 4)

d. Pool layers (2, 3, 4)

2. Fixing other parameters to see best Activation (3x3, 1, 2 maxpool)

Activation	Relu	Tanh
Train-Acc	0.994	0.991
Test-Acc	0.991	0.986
Graph	Relu_act	Tanh_act

3. Fixing other parameters to see best Filter size and stride (relu, 2 maxpool)

Size	3x3			5x5			7x7		
Stride	1	2	3	1	2	3	1	2	3
Train-Acc	0.994	0.971	0.969	0.994	0.988	0.981	0.991	0.978	0.983
Test-Acc	0.988	0.97	0.965	0.991	0.984	0.98	0.986	0.972	0.981
Graph	fig 1	fig 2	fig 3	fig 4	fig 5	fig 6	fig 7	fig 8	fig 9

Observation:

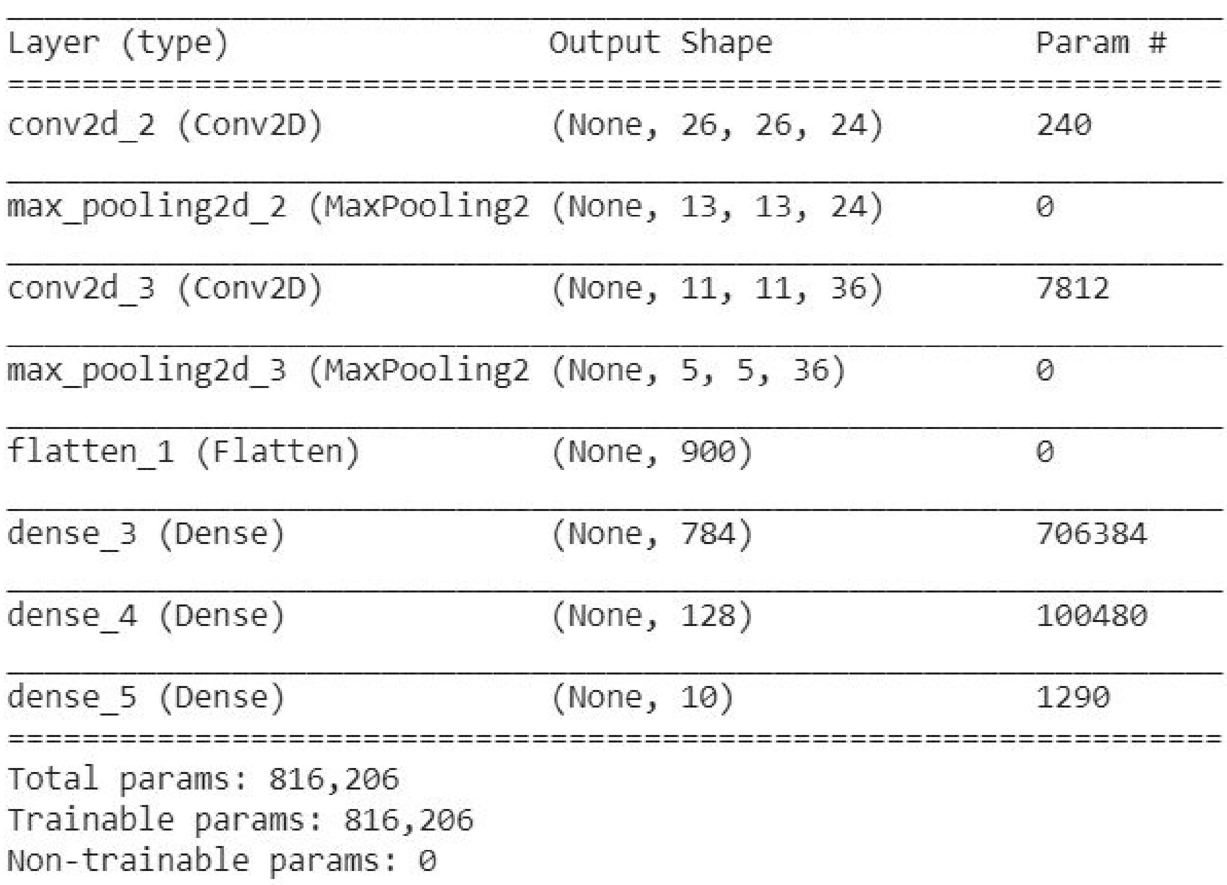
Decreasing the stride length always result in better accuracy and the opposite happens with the filter size, as size increase the accuracy increases till an overfitting happens

4. Fixing other parameters to se best num of maxpool layers

Maxpool num	2	3	4
Train-Acc	0.994	0.99/0.992	0.985
Test-Acc	0.989	0.988/0.989	0.985

5. Model Analysis

Analysis is done for one Model, however it could be repeated to all similarly



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_2 (Conv2D)            (None, 26, 26, 24)        240
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 13, 13, 24)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 11, 11, 36)        7812
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 5, 5, 36)          0
_________________________________________________________________
flatten_1 (Flatten)          (None, 900)               0
_________________________________________________________________
dense_3 (Dense)              (None, 784)               706384
_________________________________________________________________
dense_4 (Dense)              (None, 128)               100480
_________________________________________________________________
dense_5 (Dense)              (None, 10)                1290
=================================================================
Total params: 816,206
Trainable params: 816,206
Non-trainable params: 0
```

a. Running time = 26 sec
b. Number of parameters = 816,206
c. Number of multiplications
    i. Training = 60,000 * ((26*26*24*3*3*1) + (0) + (11*11*36*3*3*24) + (0) + (784*900) + (784*128) + (128*10)) * 5
    ii. Testing = 10,000 * ((26*26*24*3*3*1) + (0) + (11*11*36*3*3*24) + (0) + (784*900) + (784*128) + (128*10))

6. Comparison to FC NN

Model	FC-NN	CNN
Train Accuracy	0.982	0.994

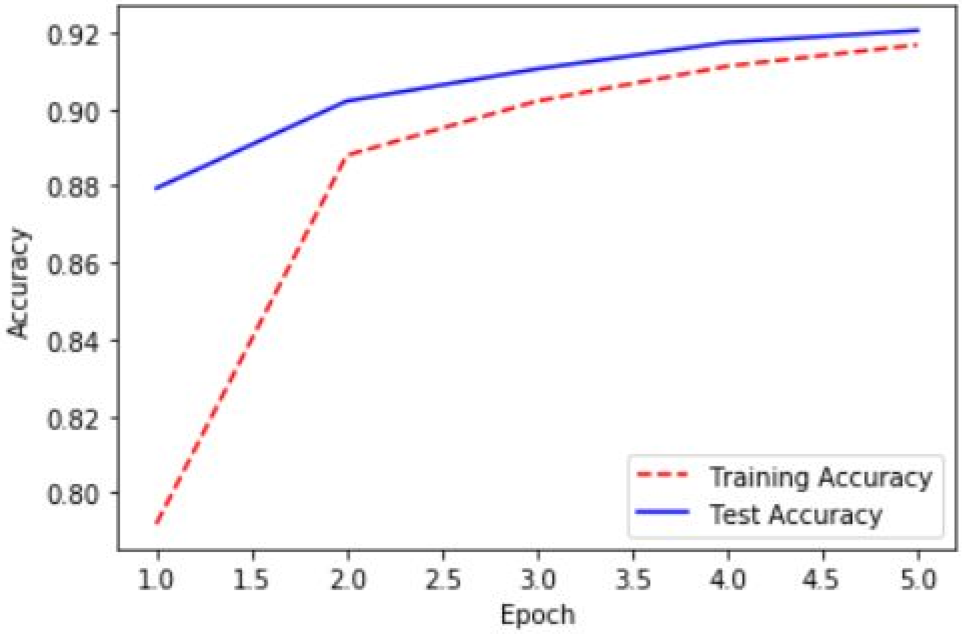
Test Accuracy	0.974	0.991
Time	23.46 sec	26.0 sec

Appendix

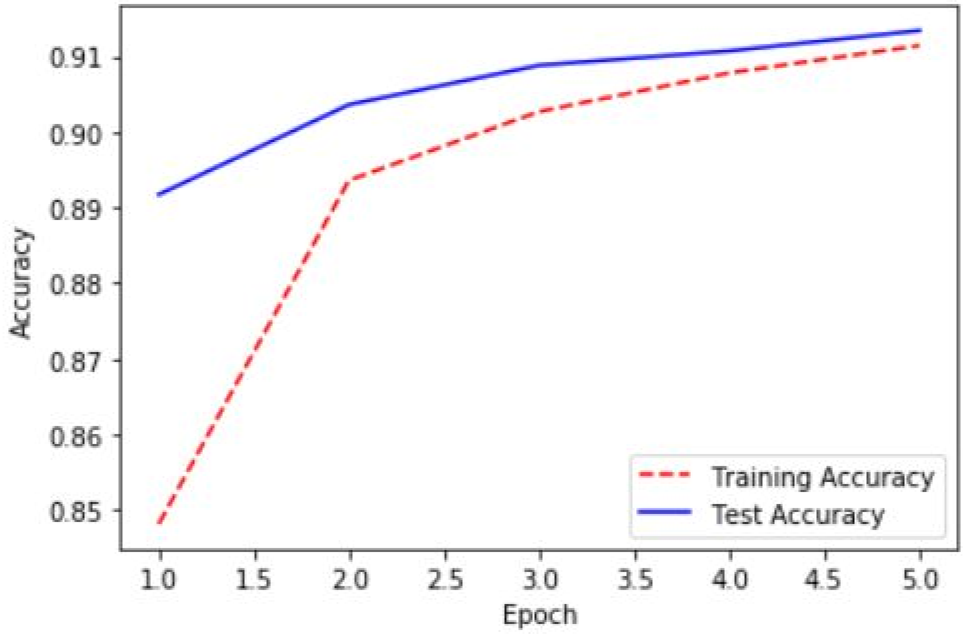
1. FC NN

a. Optimizers

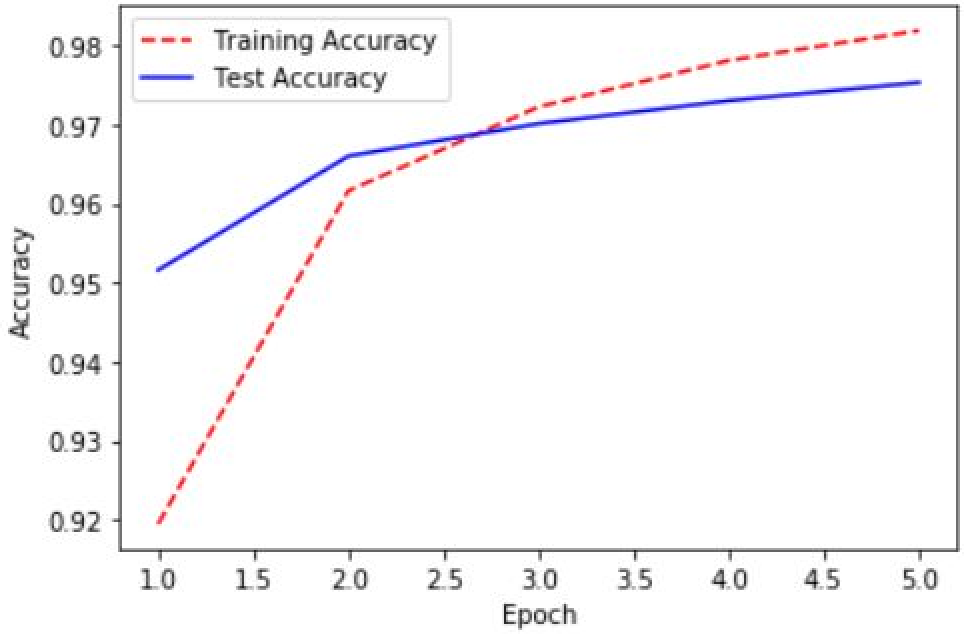
i. Sgd



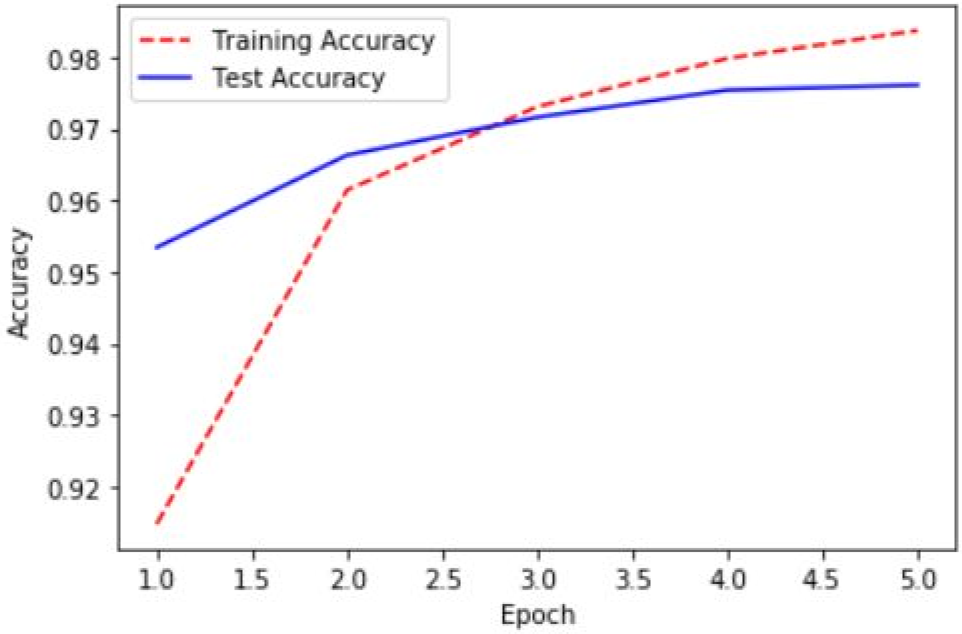
ii. Adagrad



iii. Rmsprop

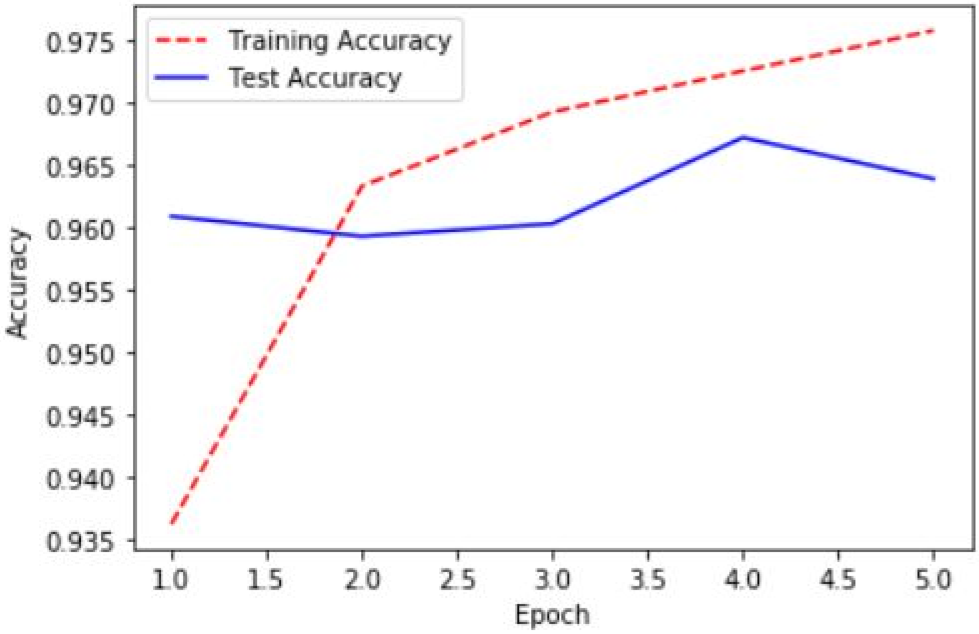


iv. Adam

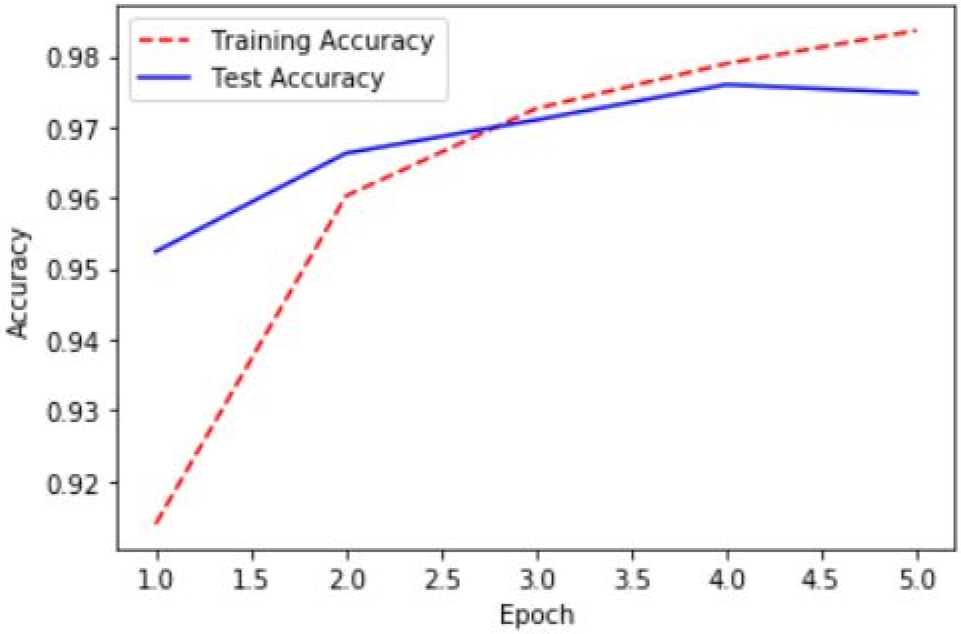


b. LR

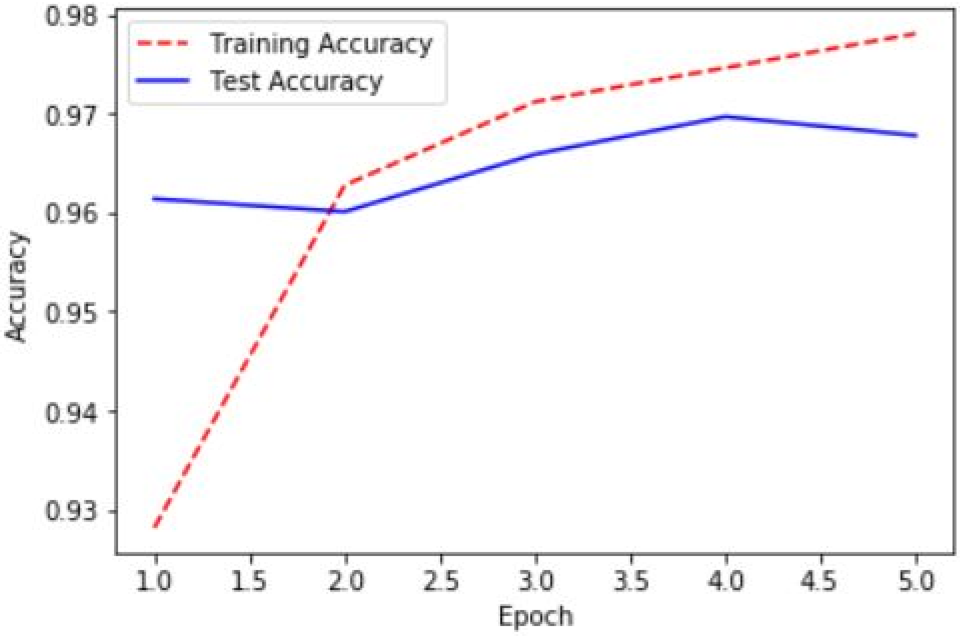
i. Adam_0.01



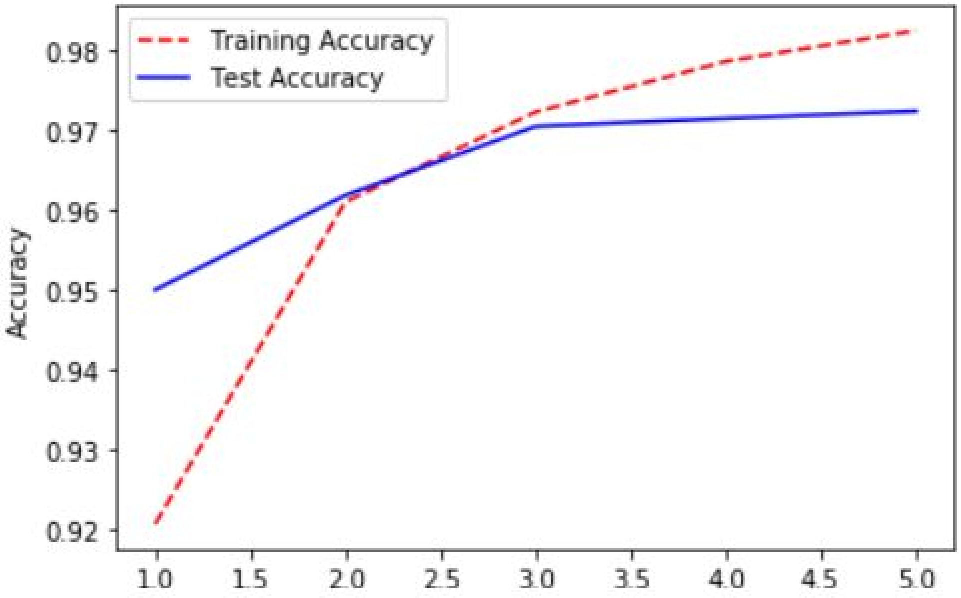
ii. Adam_0.001



iii. RMSprop_0.01

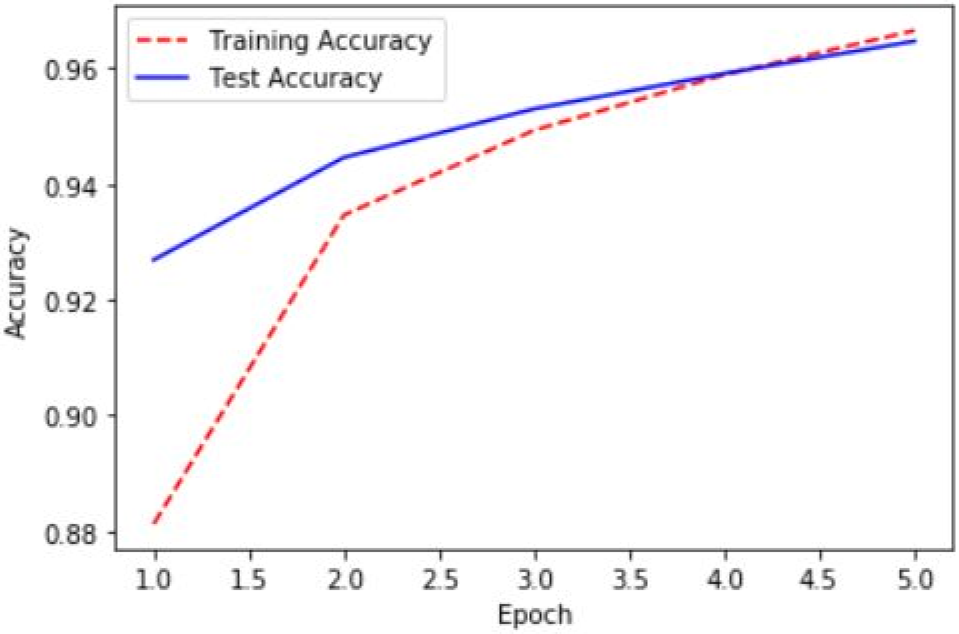


iv. RMSprop_0.001

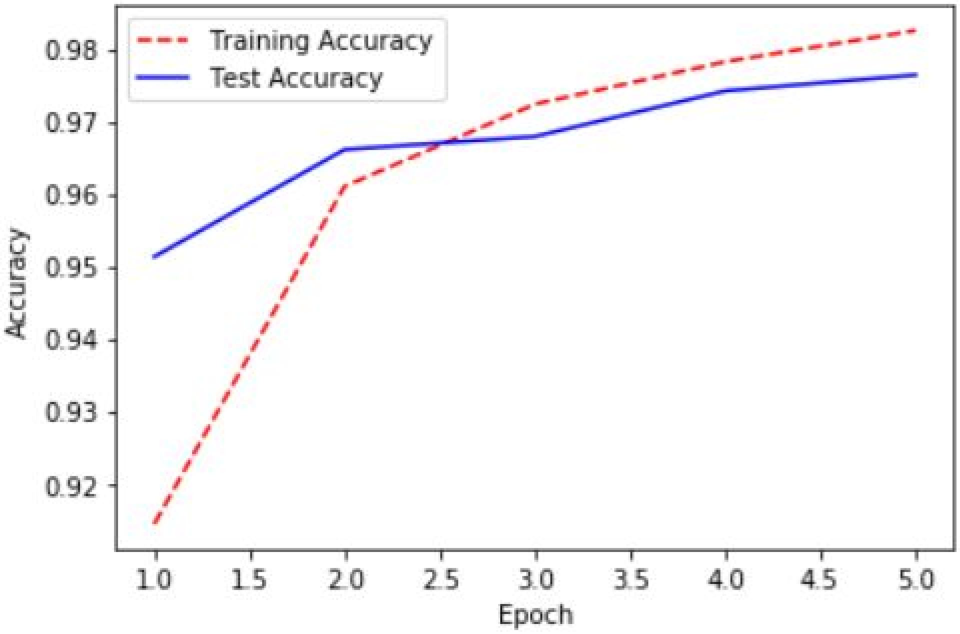


c. Activation

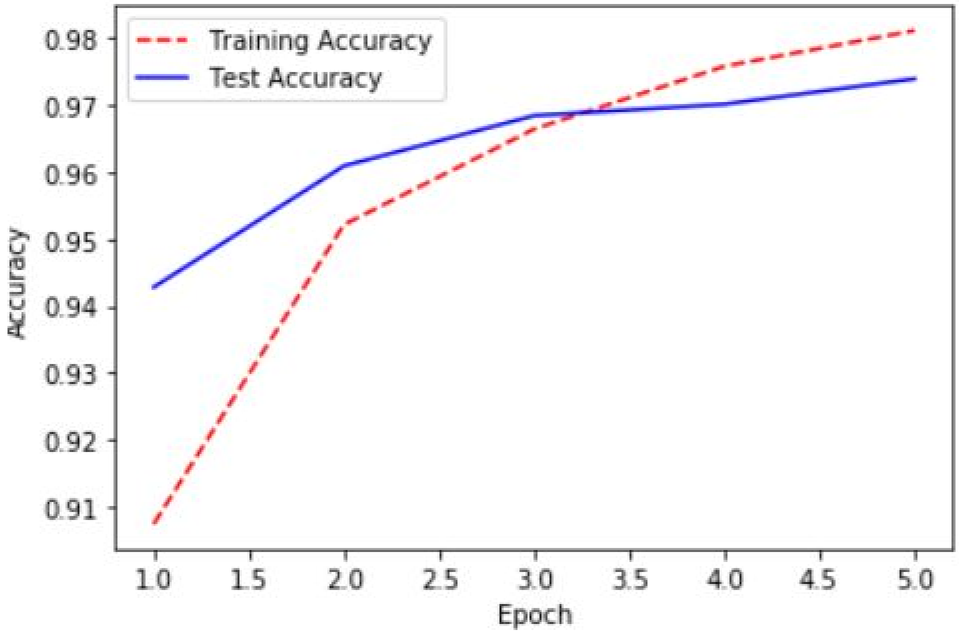
i. Sigmoid



ii. Relu



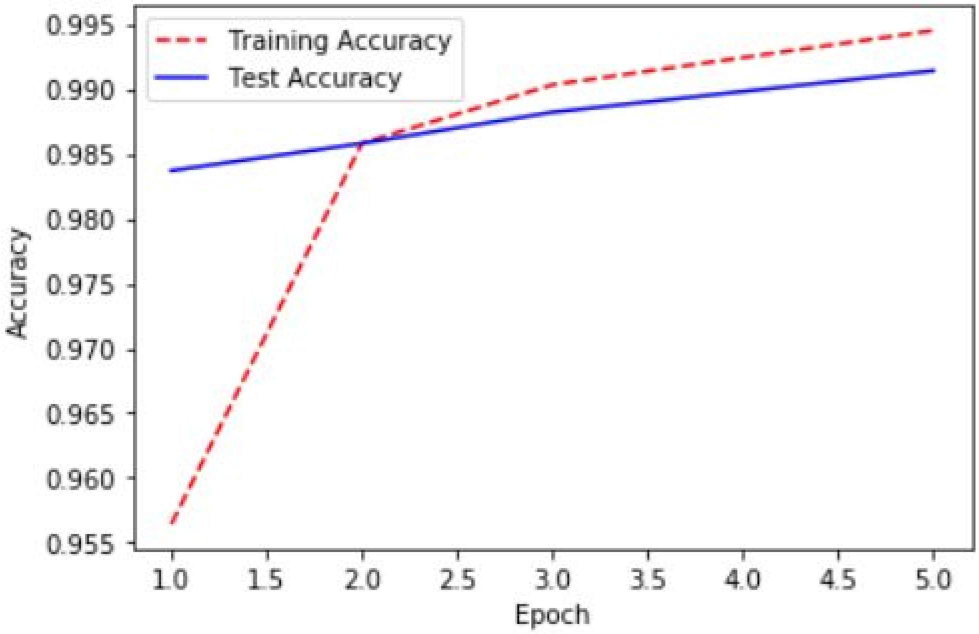
iii. Tanh



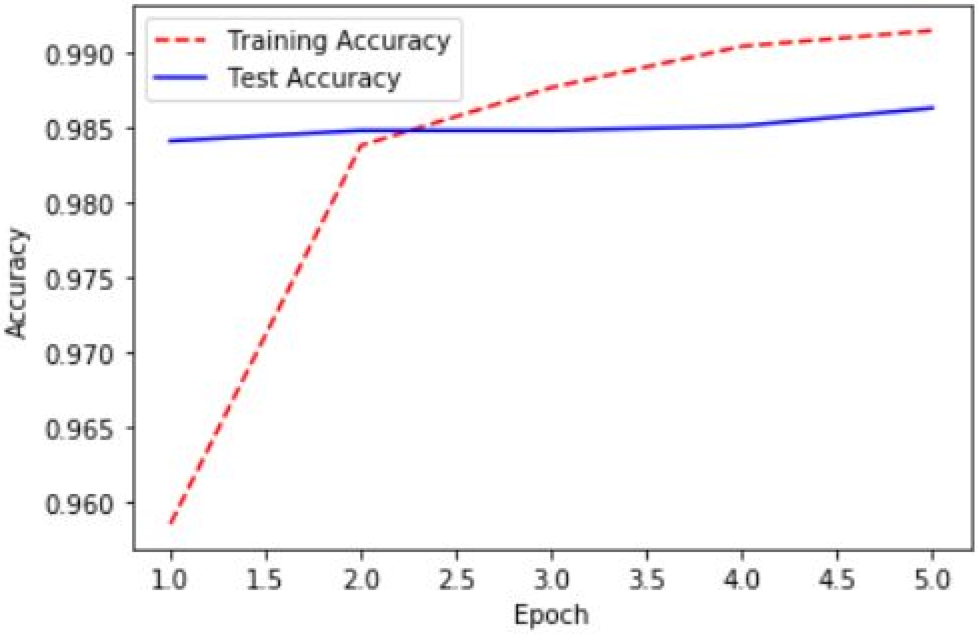
2. CNN

a. Activation

i. Relu

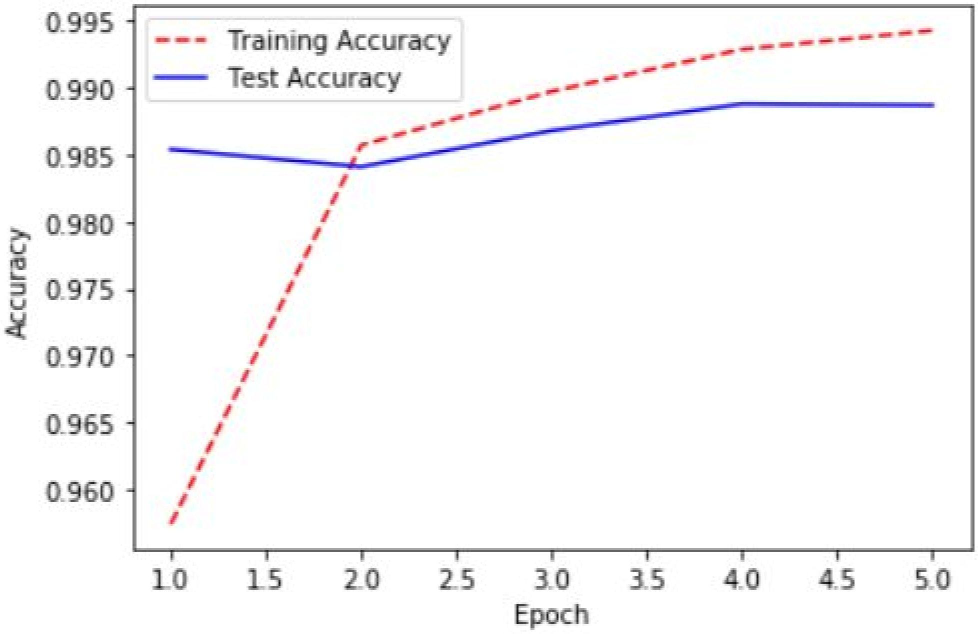


ii. Tanh

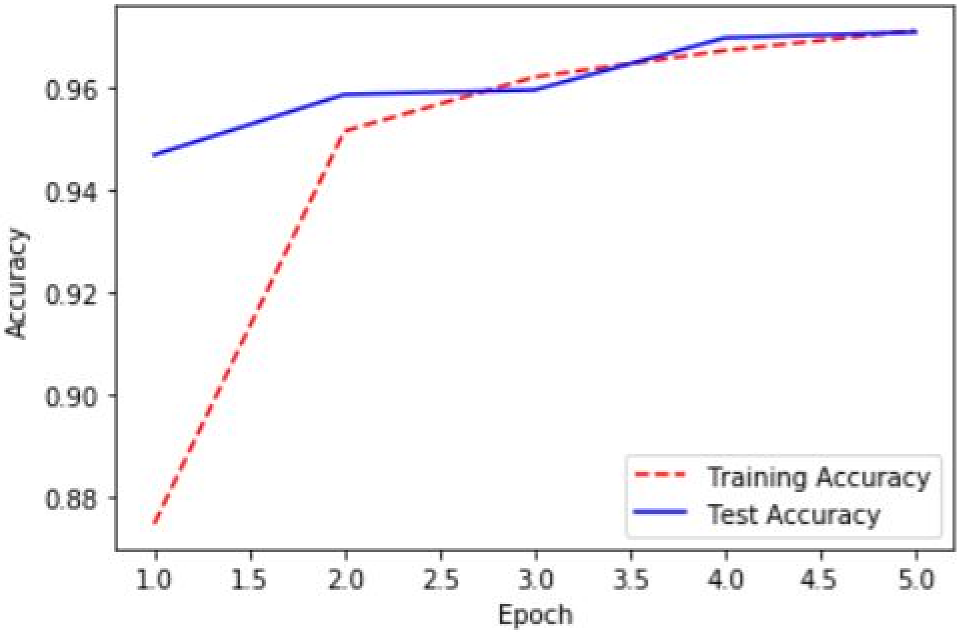


b. Filter size & Stride

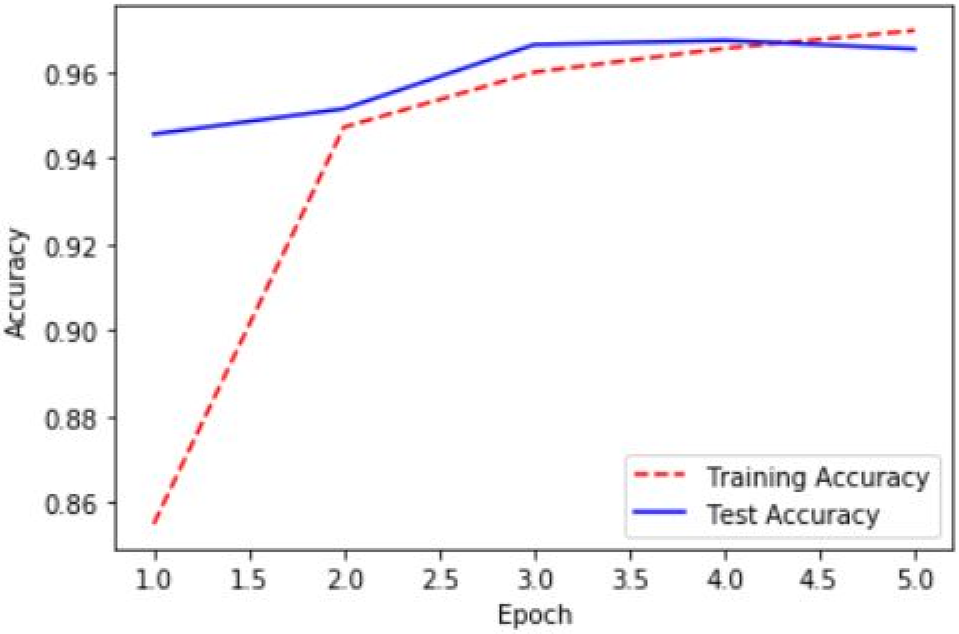
i. 3x3_1



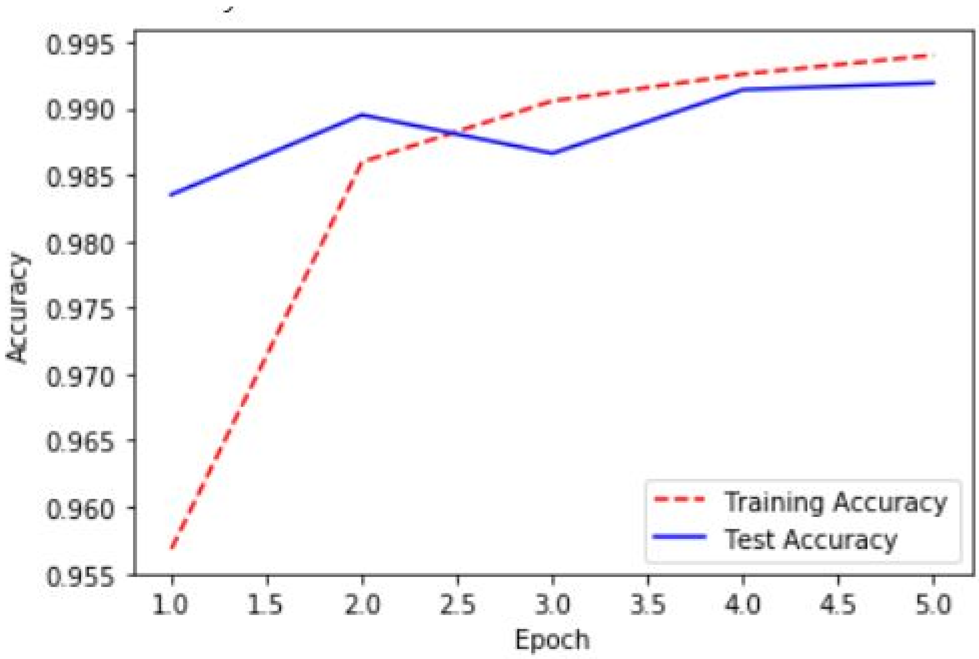
ii. 3x3_2



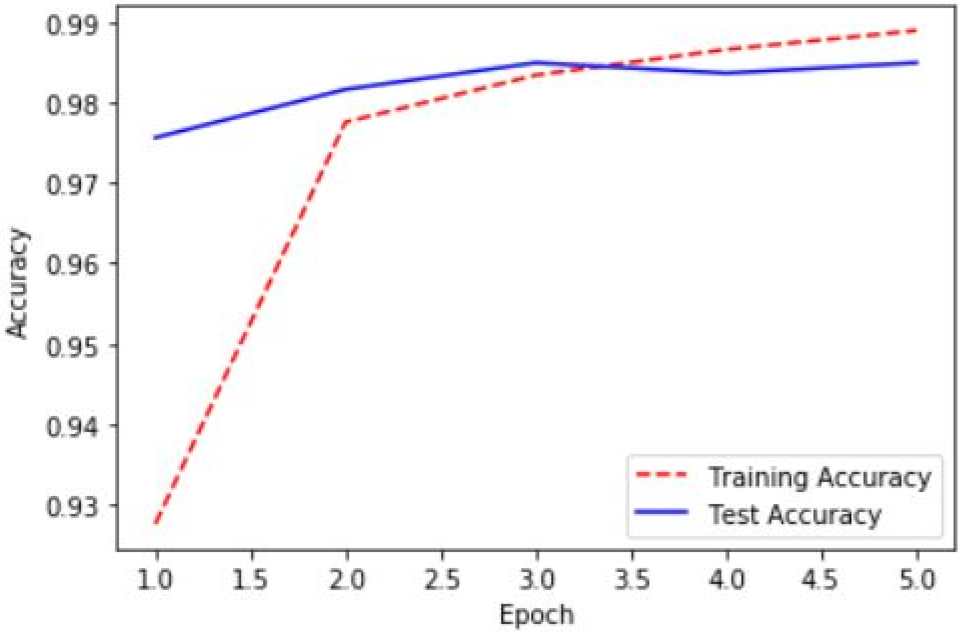
iii. 3x3_3



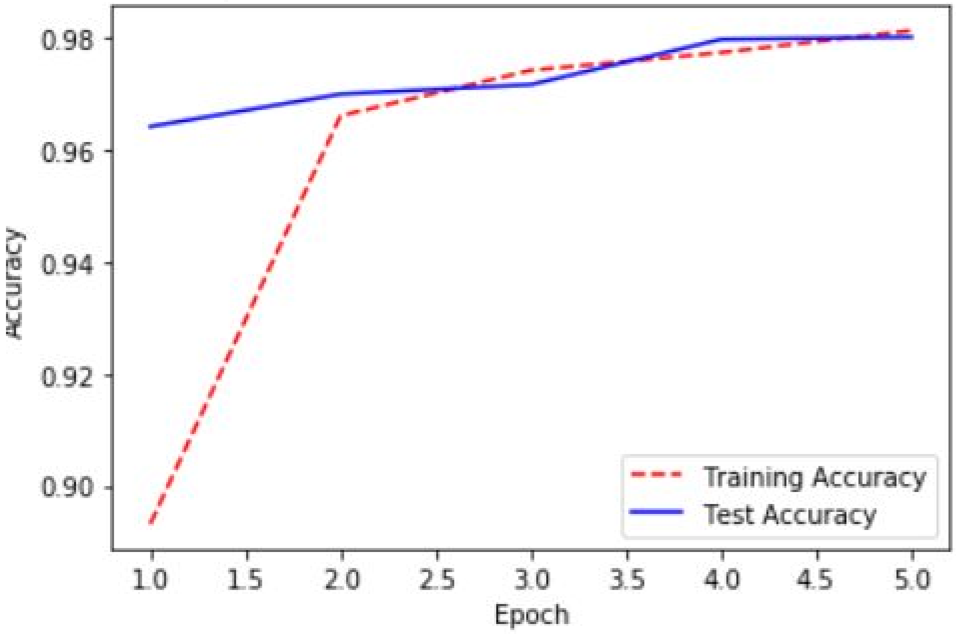
iv. 5x5_1



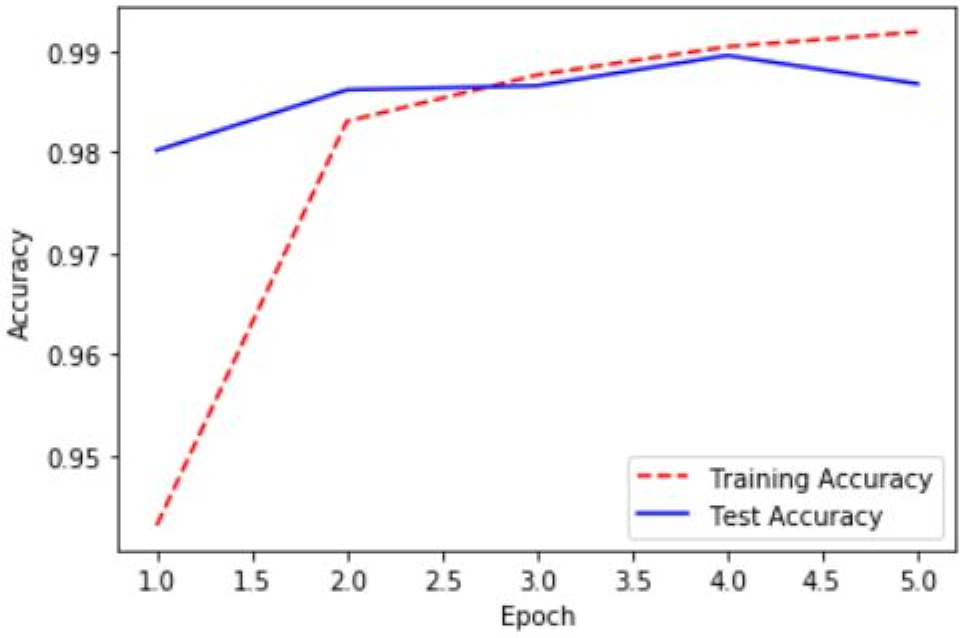
v. 5x5_2



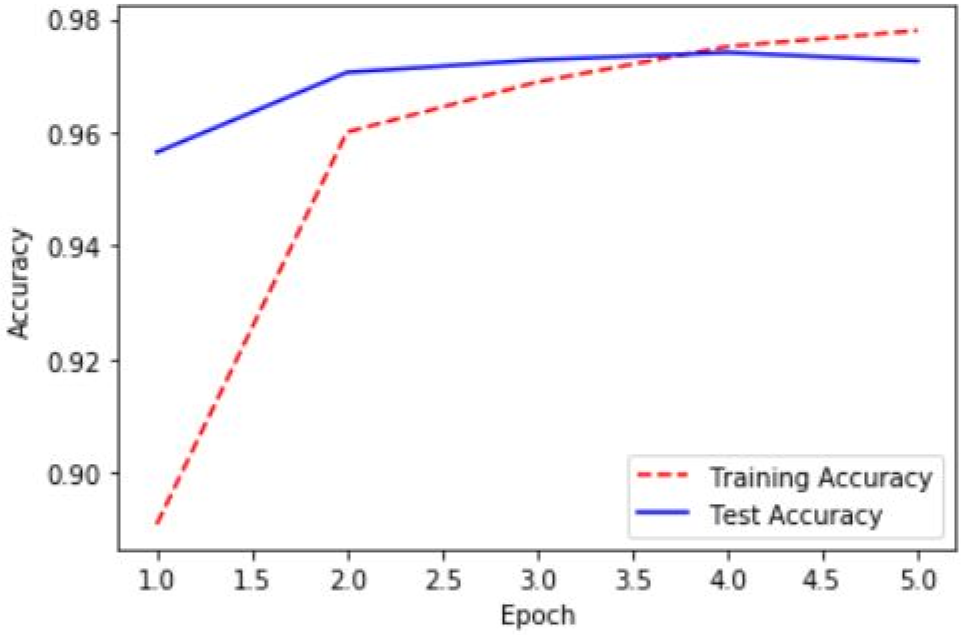
vi. 5x5_3



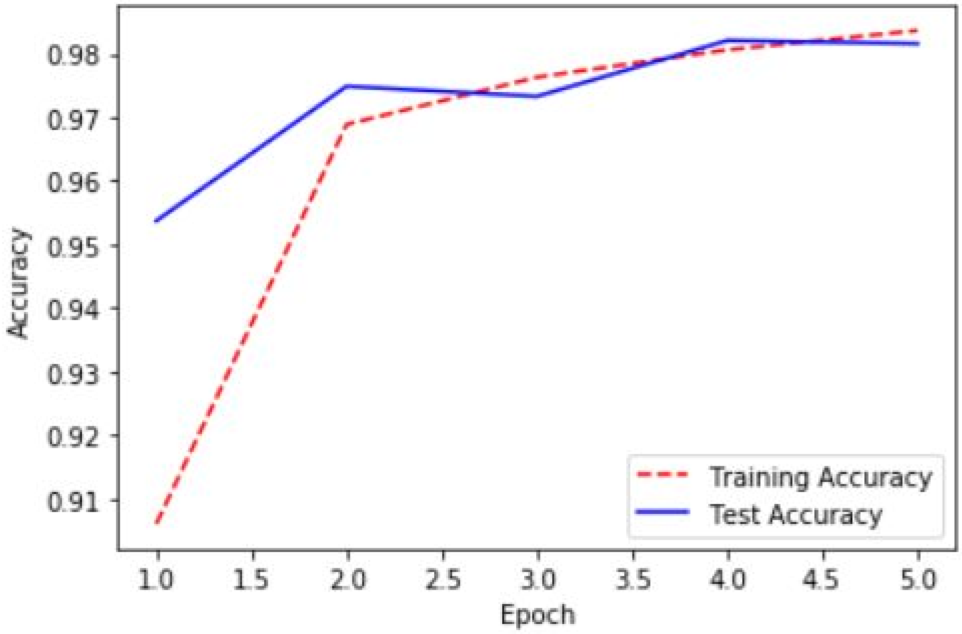
vii. 7x7_1



viii. 7x7_2



ix. 7x7_3


0.98
0.97
0.96
0.95
0.94
0.93
0.92
0.91
Accuracy
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
Epoch
Training Accuracy
Test Accuracy

c. .