Batch Size : 256

train ratio : 0.85

To run this file , please select run all button.

Model_1:

```
fnn = [
    Dense_Layer(784,1024),
    Batch_Normalization(1024),
    activationFunction_ReLU(),
    Dense_Layer(1024,10),
    Soft_Max()
]
```

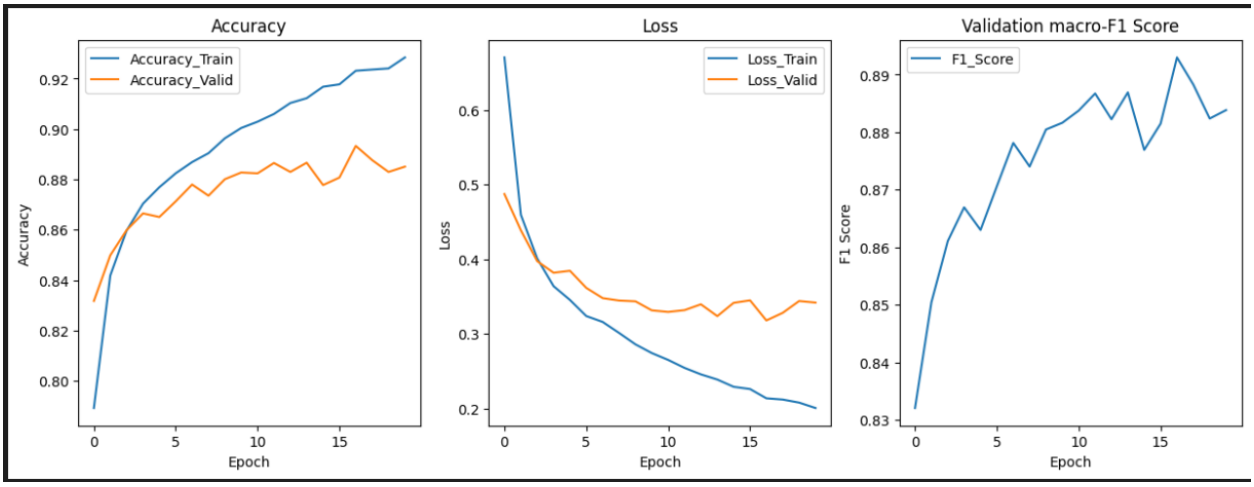
0.1s

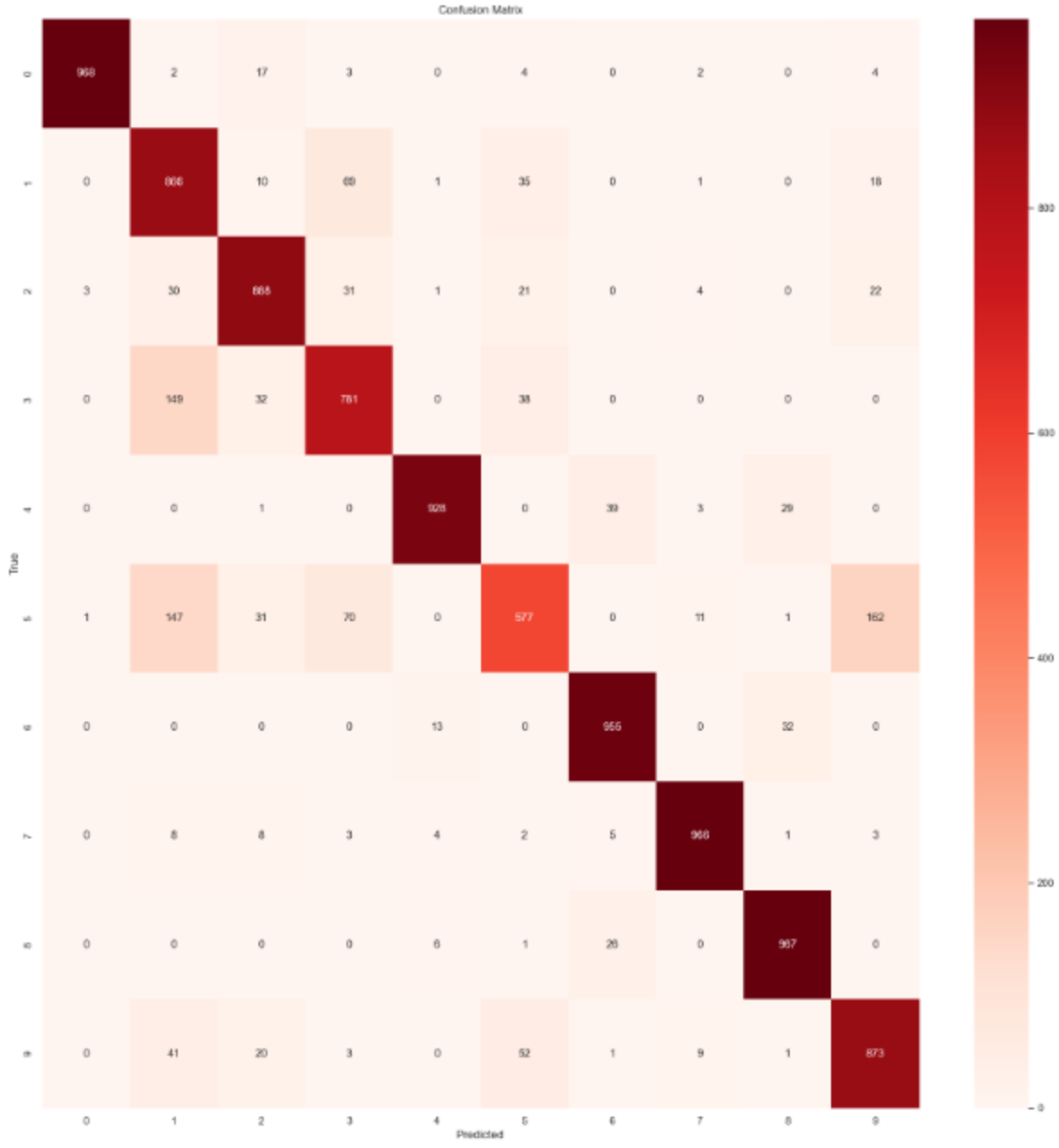
Learning Rate: 0.001

Best valid F1 value = 0.89

It is the best model , i have got from all kind of model

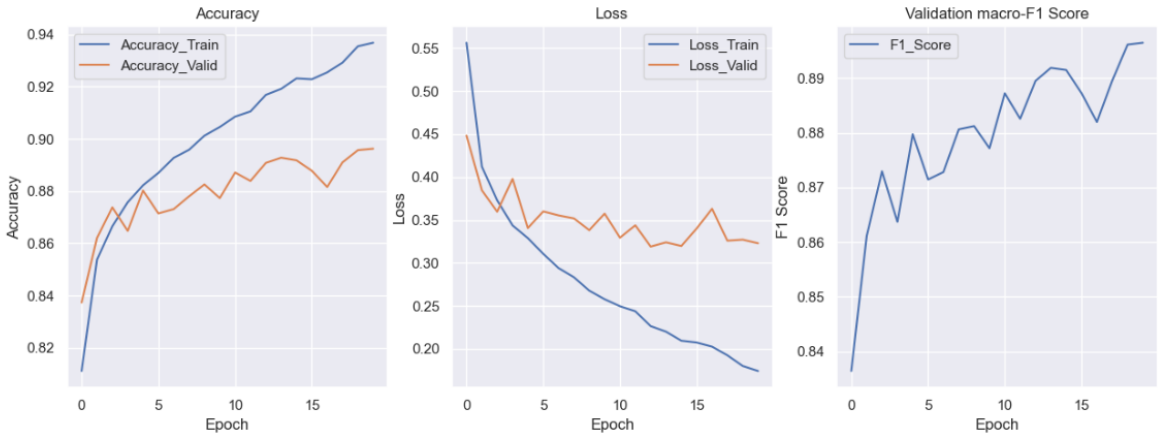
Test Loss: 0.376, Accuracy: 0.877, F1: 0.875

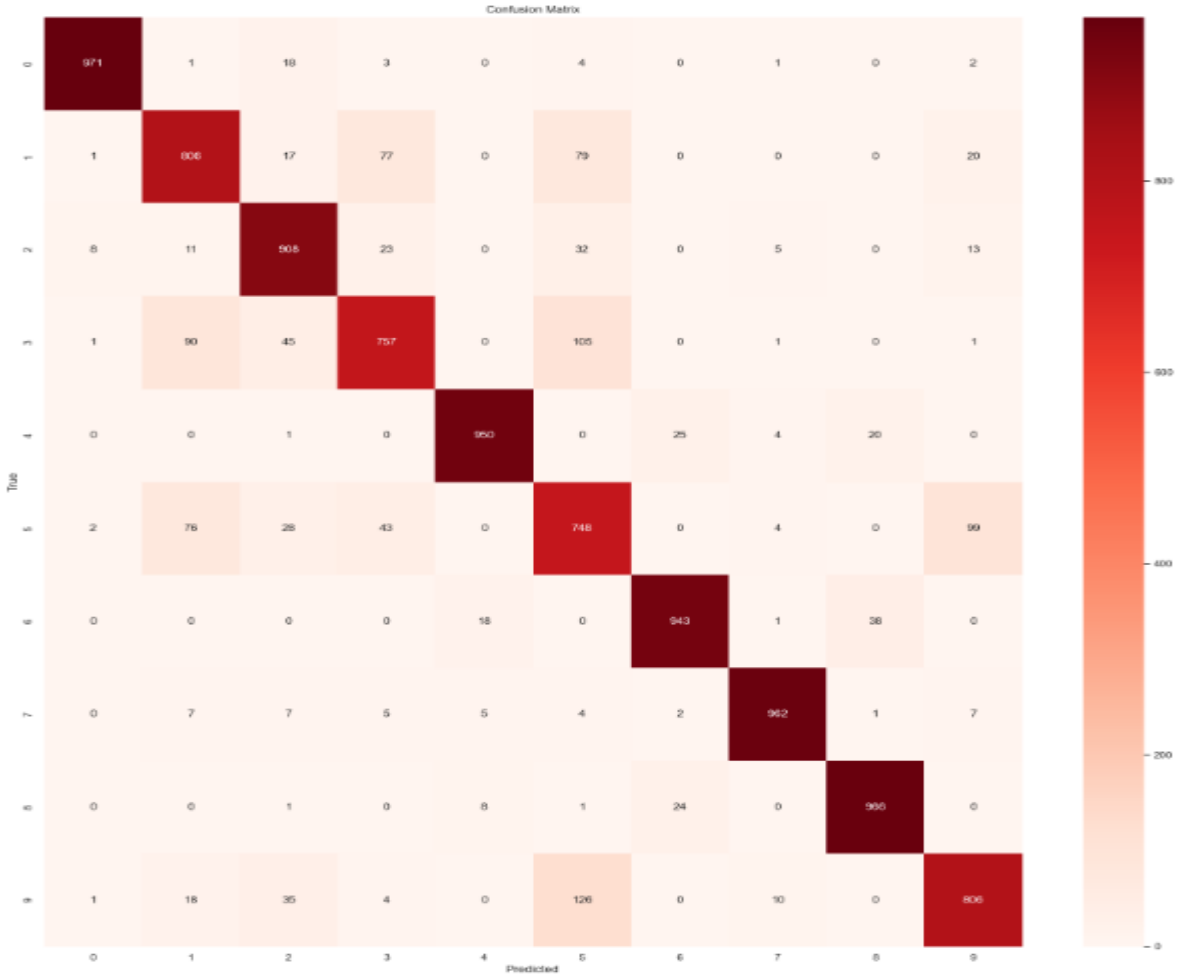



Confusion Matrix
True
Predicted
0 1 2 3 4 5 6 7 8 9
968 2 17 3 0 4 0 2 0 4
0 808 10 69 1 35 0 1 0 18
3 30 885 31 1 21 0 4 0 22
0 149 32 781 0 38 0 0 0 0
0 0 1 0 926 0 39 3 29 0
1 147 31 70 0 577 0 11 1 162
0 0 0 0 13 0 955 0 32 0
0 8 8 3 4 2 5 966 1 3
0 0 0 0 6 1 26 0 967 0
0 41 20 3 0 52 1 9 1 873
800
600
400
200
0

Learning Rate: 0.003

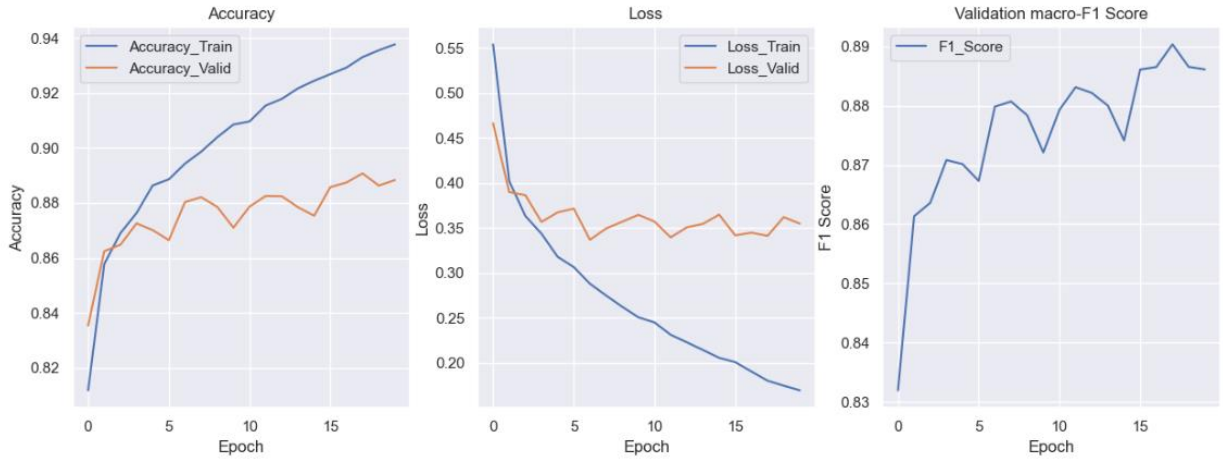
Test Loss: 0.364, Accuracy: 0.882, F1: 0.882

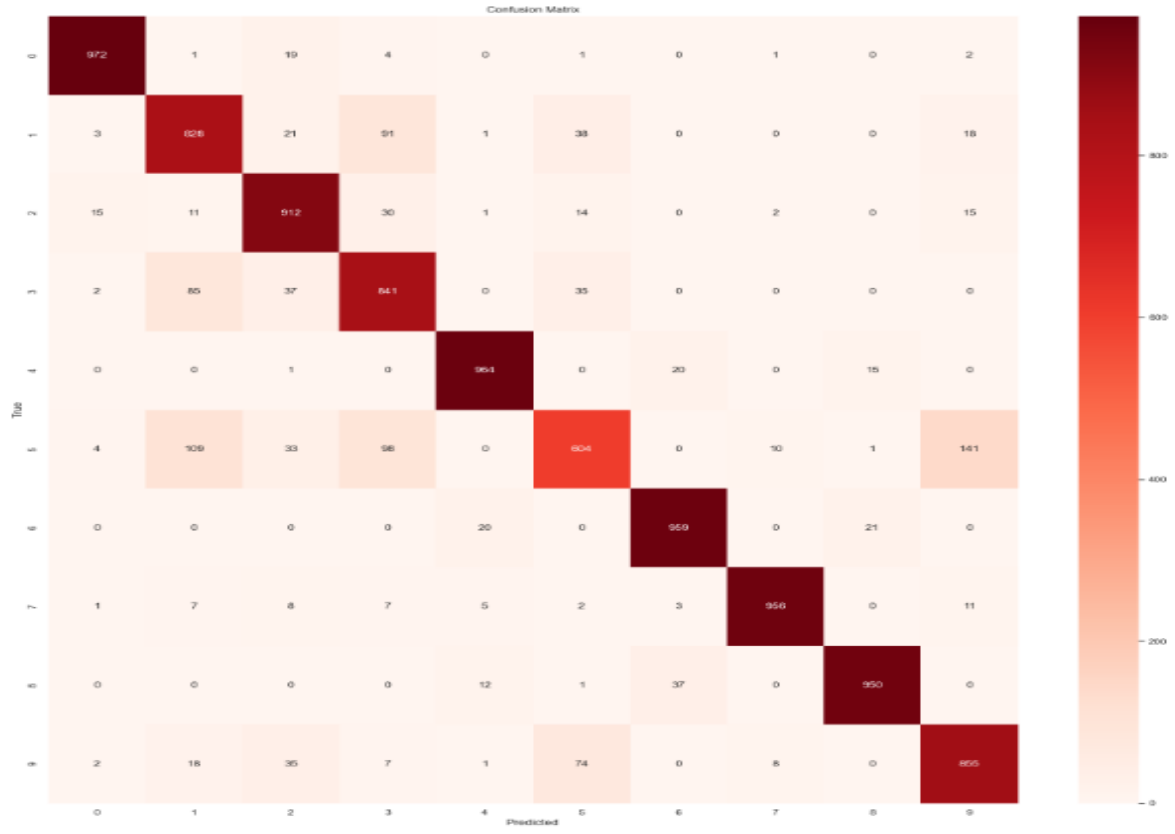




Learning Rate: 0.005

Test Loss: 0.384, Accuracy: 0.884, F1: 0.882





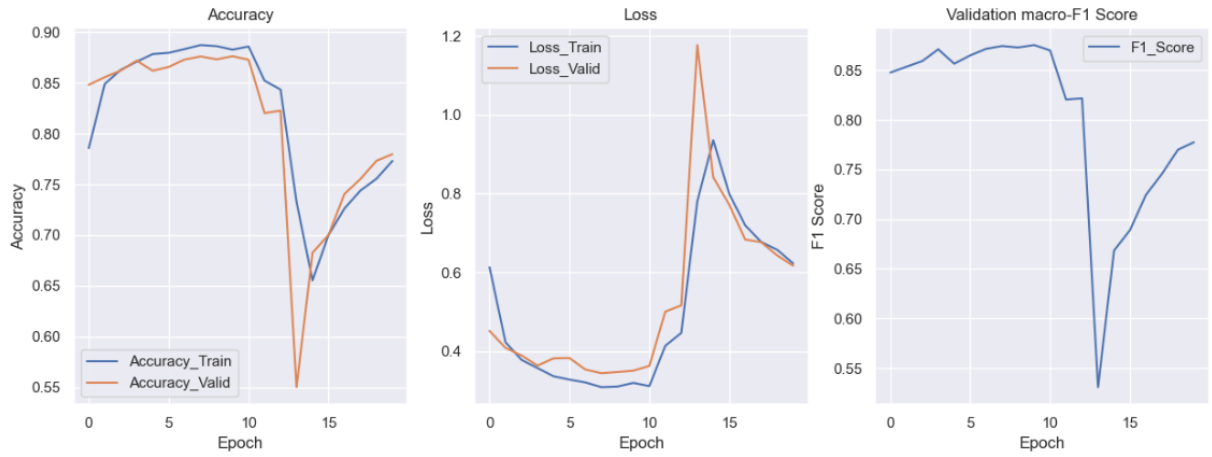
Model_2:

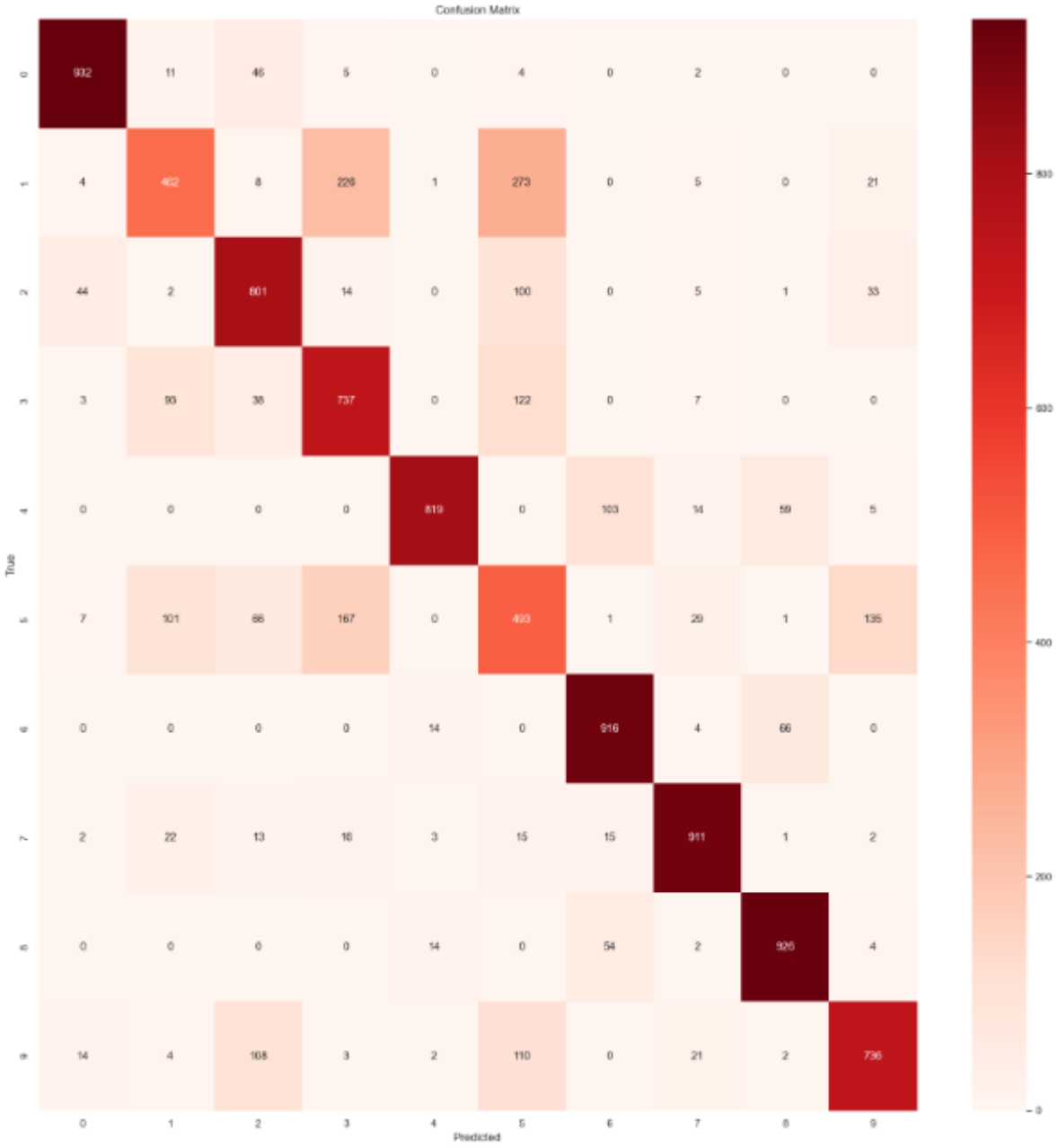
```
fnn = [
    Dense_Layer(784,128),
    Batch_Normalization(128),
    activationFunction_ReLU(),
    Dense_Layer(128,64),
    Batch_Normalization(64),
    activationFunction_ReLU(),

    Dense_Layer(64,10),
    Soft_Max()
]
```

Learning Rate : 0.005

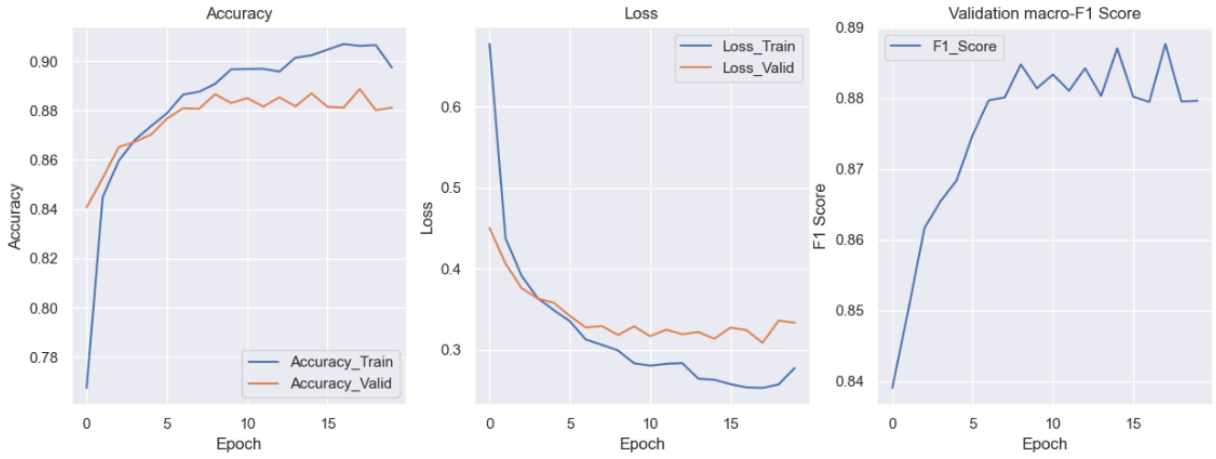
Test Loss: 0.630, Accuracy: 0.773, F1: 0.772

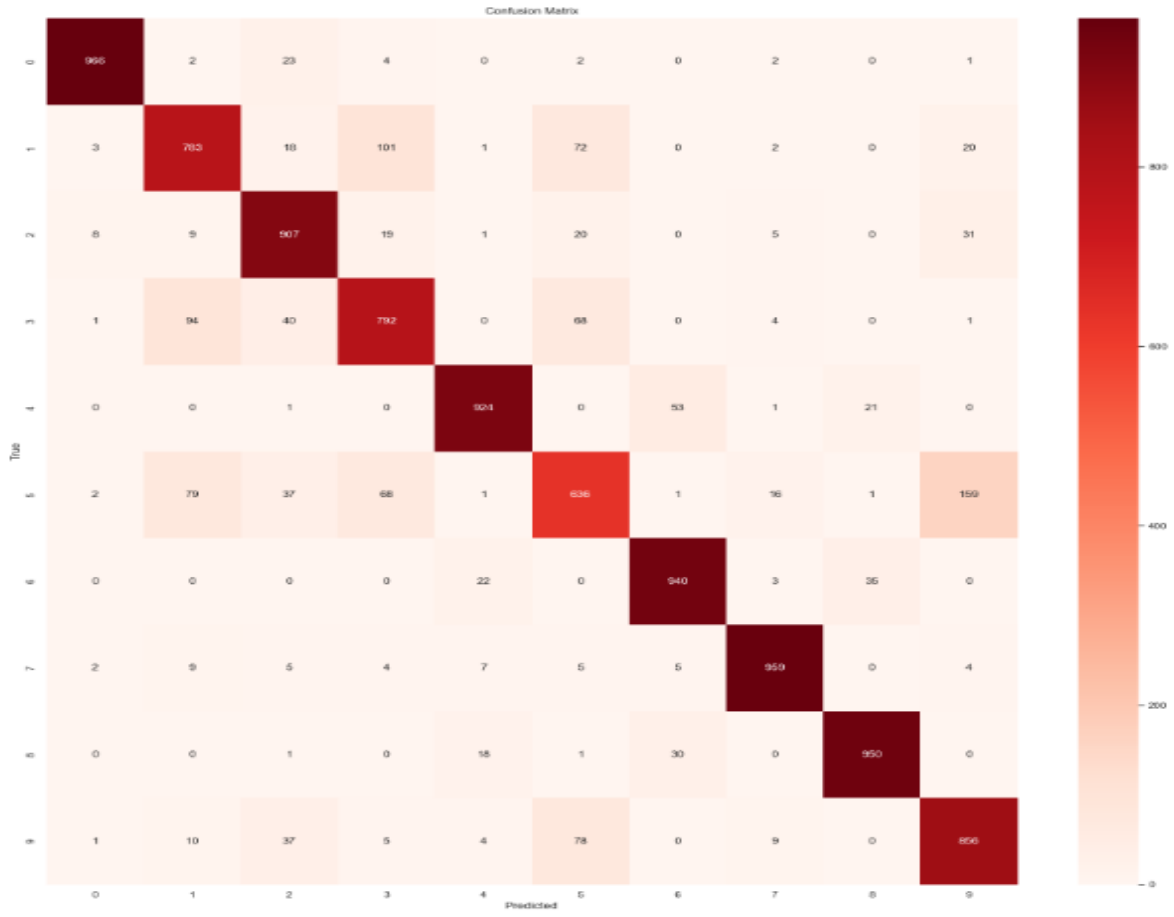



Confusion Matrix
True
Predicted
0 1 2 3 4 5 6 7 8 9
932 11 46 5 0 4 0 2 0 0
4 462 8 226 1 273 0 5 0 21
44 2 801 14 0 100 0 5 1 33
3 93 38 737 0 122 0 7 0 0
0 0 0 0 819 0 103 14 59 5
7 101 66 167 0 493 1 29 1 135
0 0 0 0 14 0 916 4 66 0
2 22 13 16 3 15 15 911 1 2
0 0 0 0 14 0 54 2 926 4
14 4 108 3 2 110 0 21 2 736
800
600
400
200
0

Learning Rate : 0.003

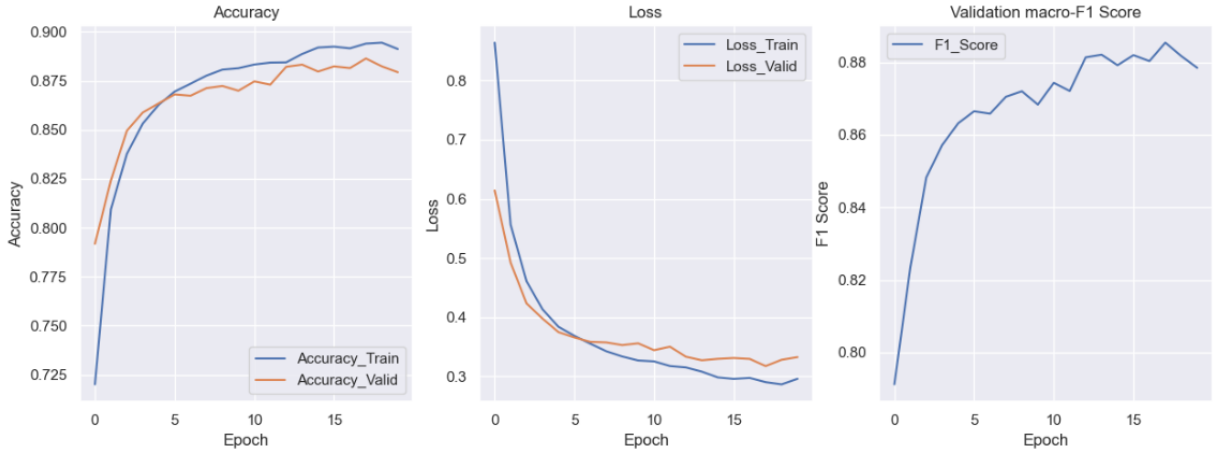
Test Loss: 0.377,Accuracy: 0.871, F1: 0.870

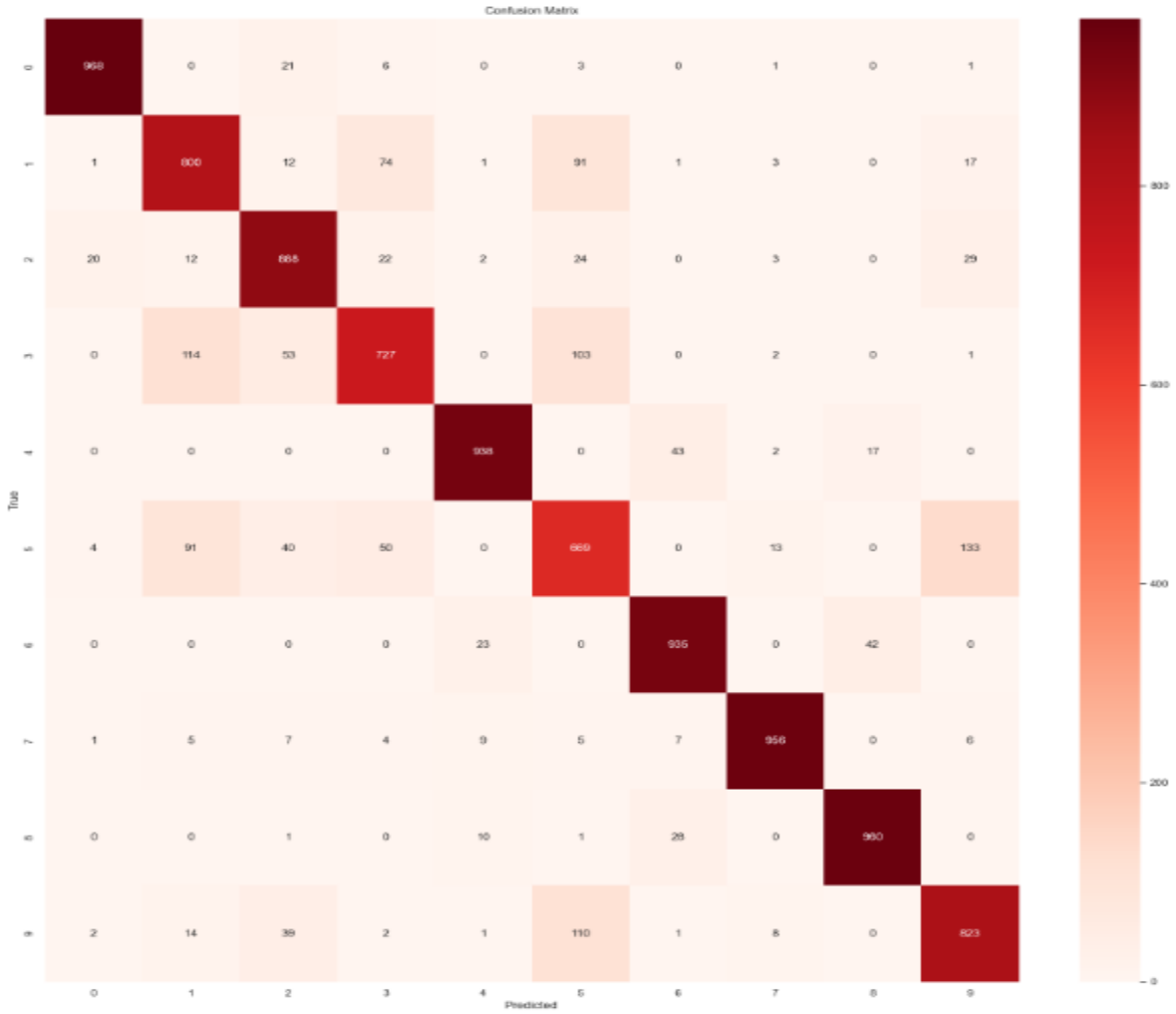




Learning Rate : 0.001

Test Loss: 0.371,Accuracy: 0.866, F1: 0.866





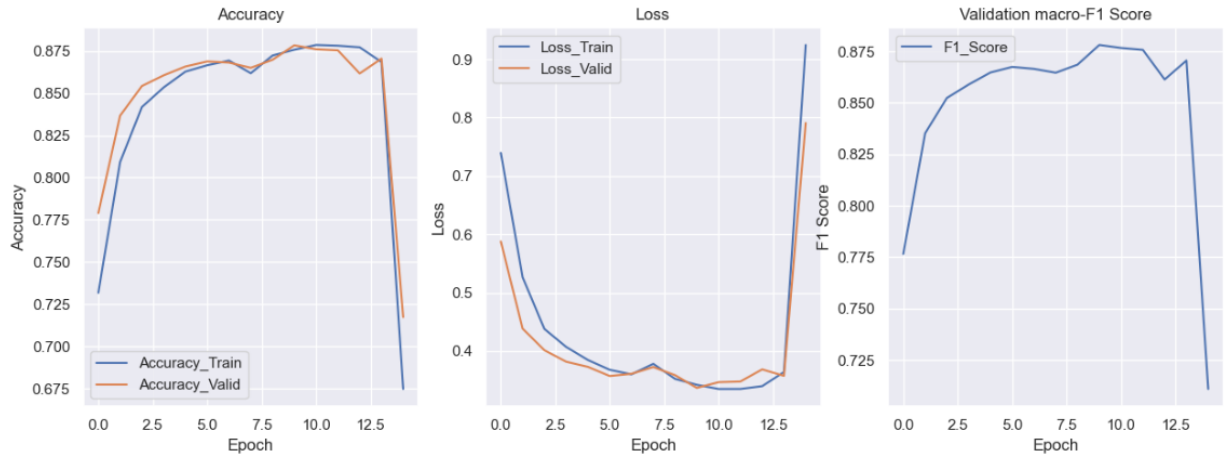
Model_3:

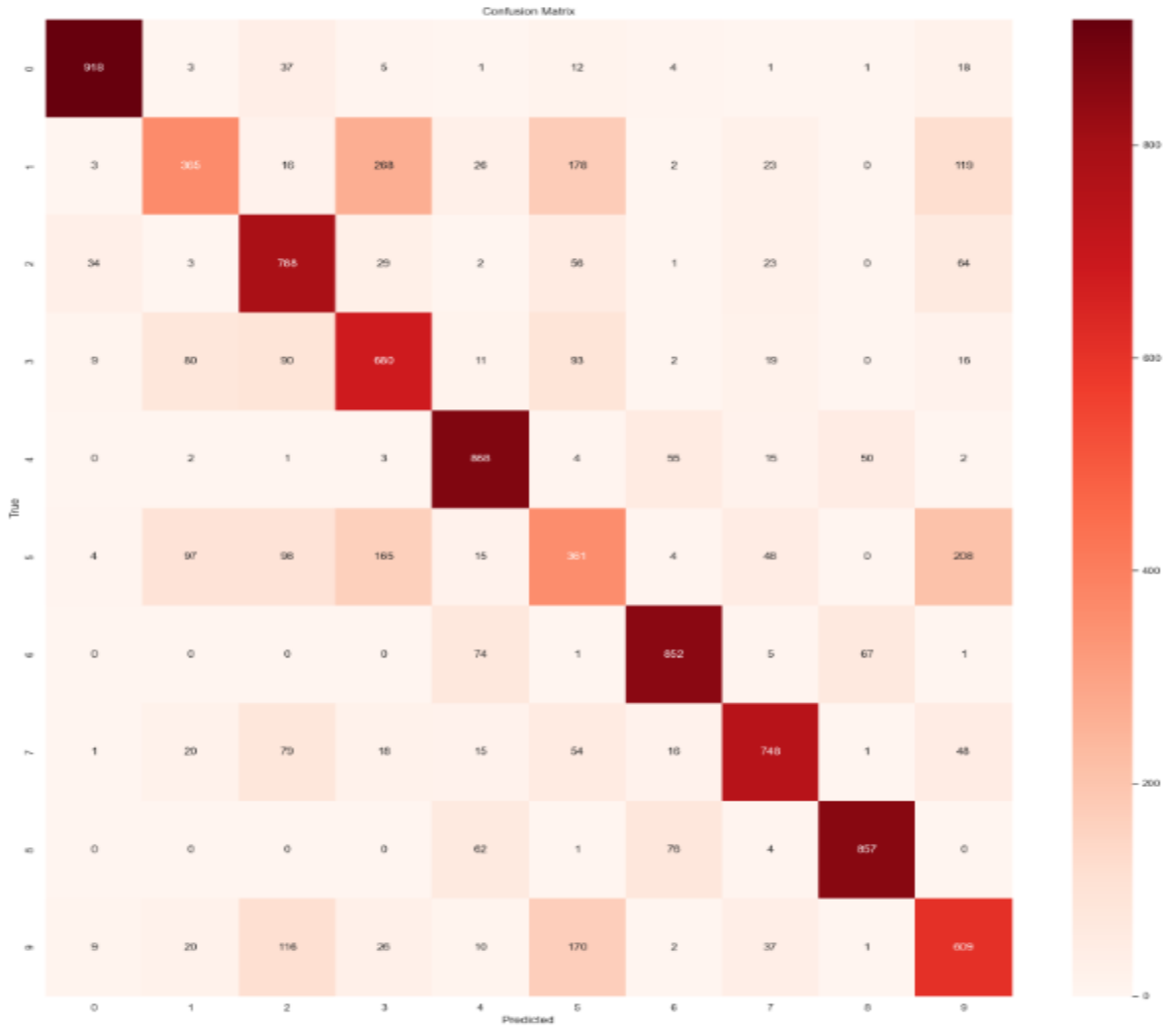
Epoch: 15

```
fnn = [
    Dense_Layer(784,1024),
    Batch_Normalization(1024),
    activationFunction_ReLU(),
    Dense_Layer(1024,256),
    Batch_Normalization(256),
    activationFunction_ReLU(),
    Drop_Out(0.5),
    Dense_Layer(256,64),
    Batch_Normalization(64),
    activationFunction_ReLU(),
    Dense_Layer(64,10),
    Soft_Max()
]
```

Learning Rate : 0.005

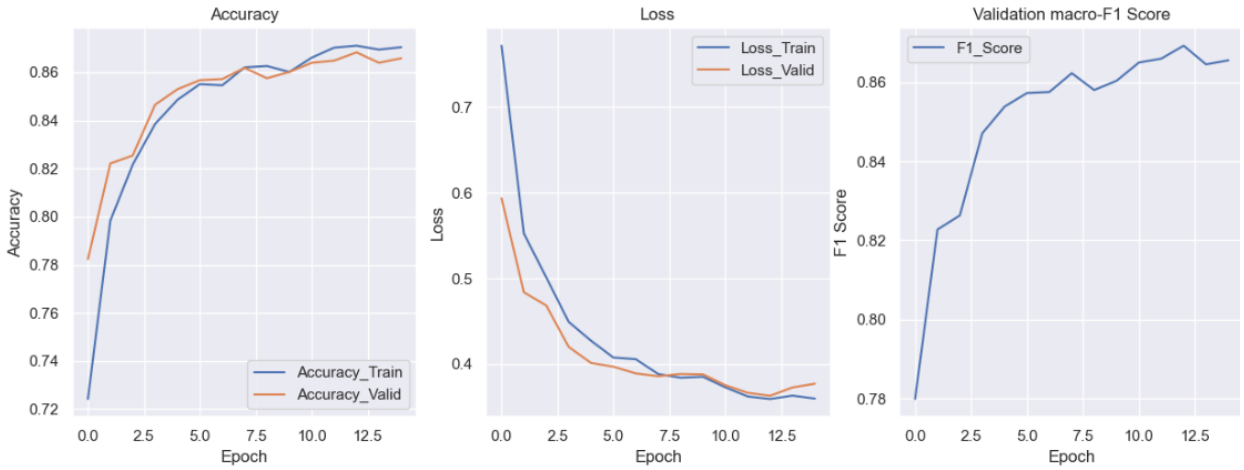
Test Loss: 0.823,Accuracy: 0.705, F1: 0.700

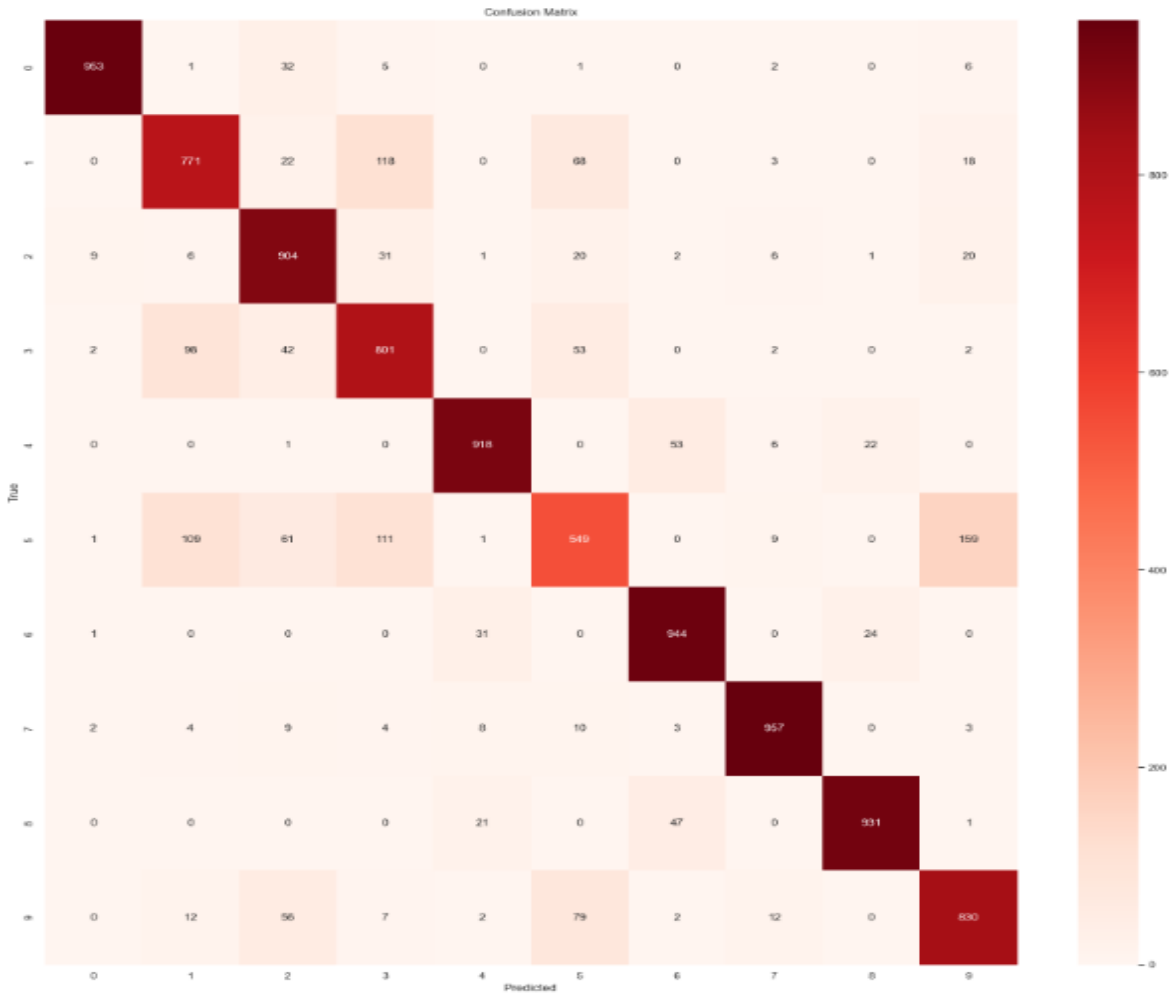

Accuracy
Accuracy
Epoch
Accuracy_Train
Accuracy_Valid
Loss
Loss
Epoch
Loss_Train
Loss_Valid
Validation macro-F1 Score
F1 Score
Epoch
F1_Score


Confusion Matrix
True
Predicted

Learning Rate : 0.003

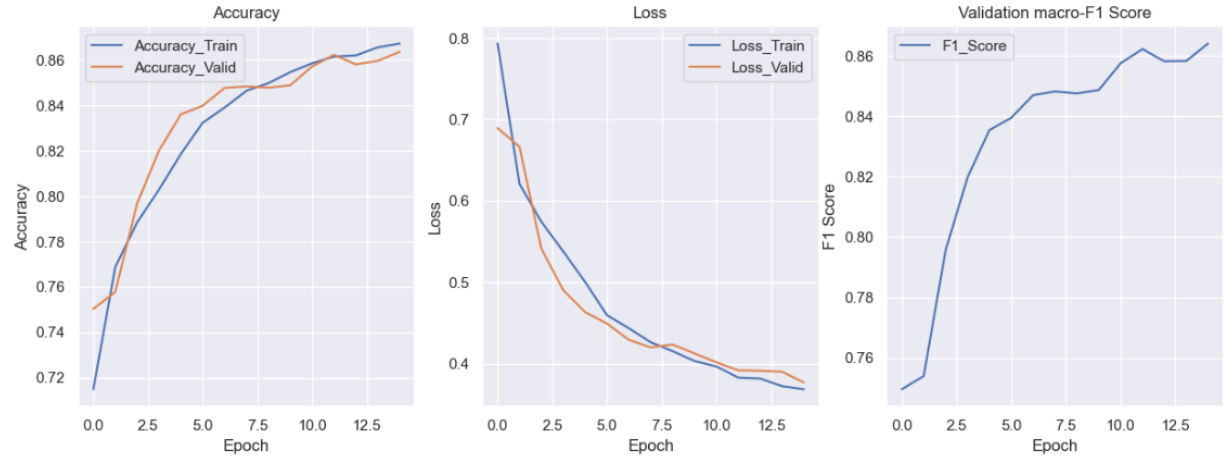
Test Loss: 0.414, Accuracy: 0.856, F1: 0.854

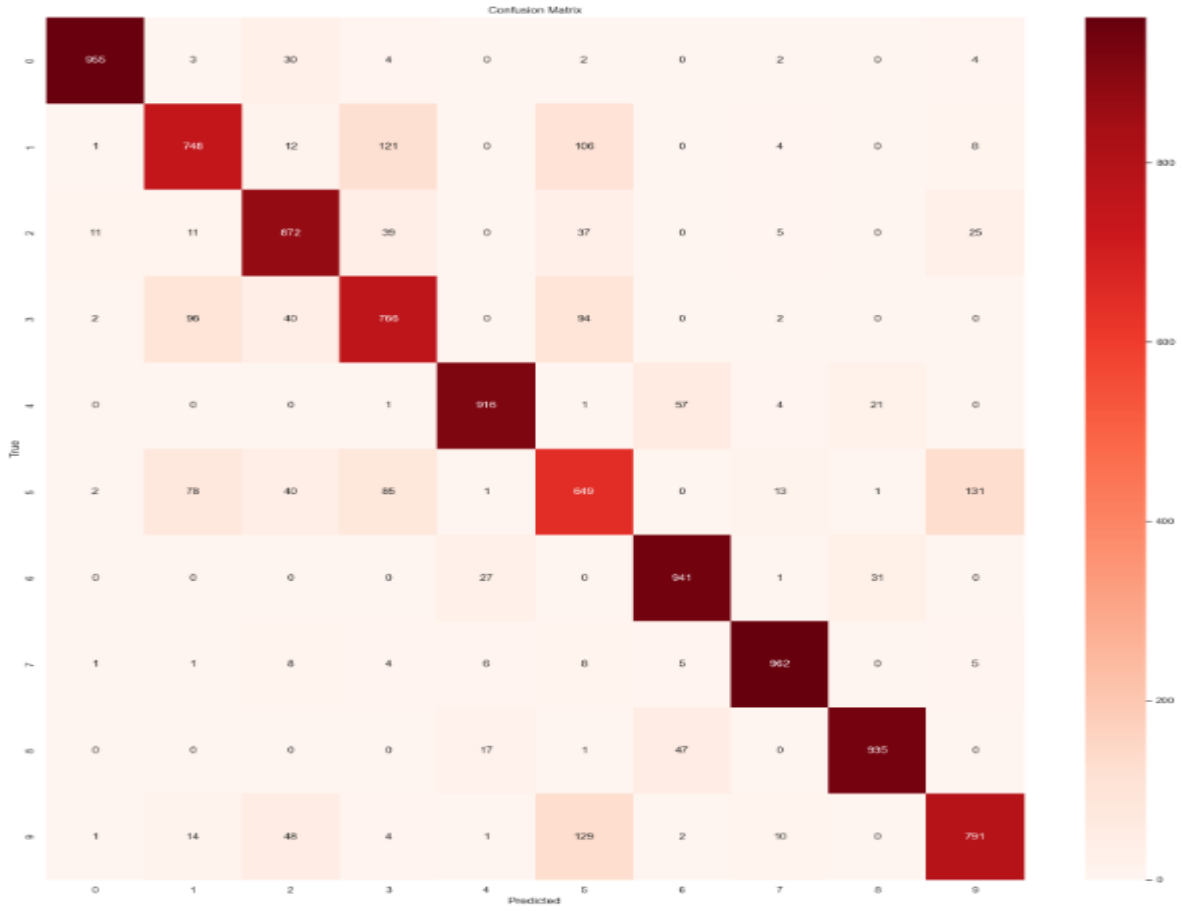




Learning Rate : 0.002

Test Loss: 0.409,Accuracy: 0.854, F1: 0.854





Learning Rate : 0.001

Test Loss: 0.451,Accuracy: 0.837, F1: 0.836