LT2326/LT2926

Assignment 1 - Thai & English OCR

name: Daniilidou Viktoria Paraskevi

How to run the scripts:

1. Splitting the dataset & challenges

In order to get the splitting of the data you have to run on the server this command

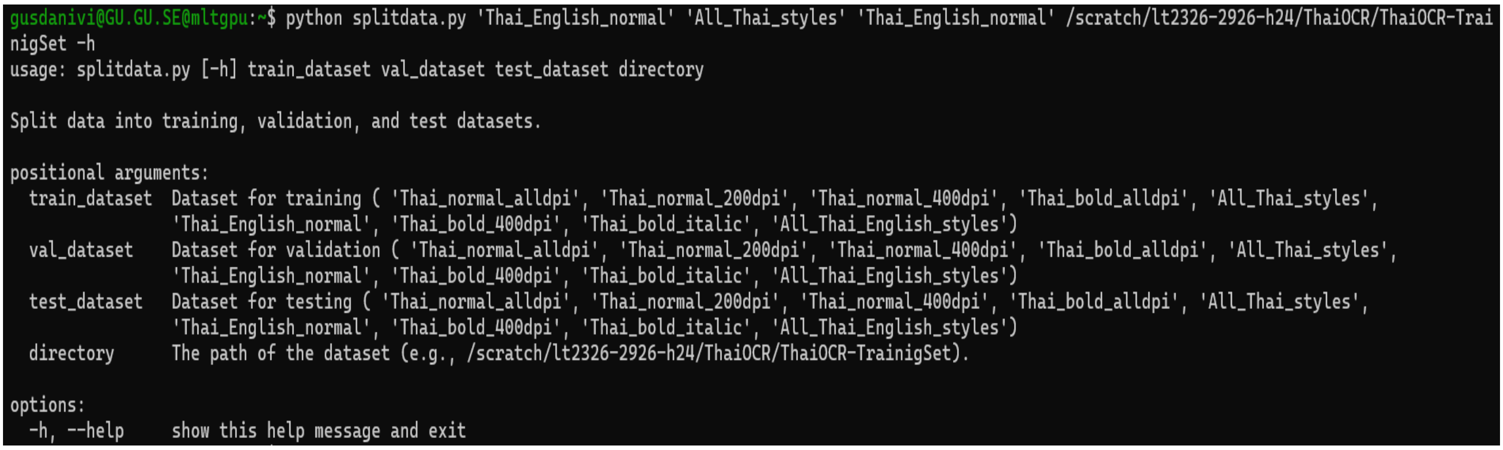
python splitdata.py 'Thai_English_normal' 'All_Thai_styles' 'Thai_bold_400dpi' /scratch/lt2326-2926-h24/ThaiOCR/ThaiOCR-TrainigSet

in this path 'Thai_English_normal' is the training, 'All_Thai_styles' is the validation and 'Thai_bold_400dpi' is the test.

Splitting the data was challenging for two reasons. The first one was to be able to separate and call the bold italic from the bold and the second one was the difficulty to ignore and take only one language, when I wanted for example just the Thai bold 200dpi category.

Also, when I was working on this assignment in the beginning I formed my code in a jupyter notebook so in order not to change it in the script the train/validation/test split is hardcoded in the split_train_val_test() function, something that I know it's not efficient and It would be more beneficial to allow language style and dpi to be configurable from the command line. So, the datasets you can choose for training test and validation are those based on the board that was on canvas.

'Thai_normal_alldpi', 'Thai_normal_200dpi', 'Thai_normal_400dpi', 'Thai_bold_alldpi', 'All_Thai_styles', 'Thai_English_normal', 'Thai_bold_400dpi', 'Thai_bold_italic', 'All_Thai_English_styles'.



```
gusdanivi@GU.GU.SE@mltgpu:~$ python splitdata.py 'Thai_English_normal' 'All_Thai_styles' 'Thai_English_normal' /scratch/lt2326-2926-h24/ThaiOCR/ThaiOCR-Trai
nigSet -h
usage: splitdata.py [-h] train_dataset val_dataset test_dataset directory

Split data into training, validation, and test datasets.

positional arguments:
  train_dataset  Dataset for training ( 'Thai_normal_alldpi', 'Thai_normal_200dpi', 'Thai_normal_400dpi', 'Thai_bold_alldpi', 'All_Thai_styles',
                 'Thai_English_normal', 'Thai_bold_400dpi', 'Thai_bold_italic', 'All_Thai_English_styles')
  val_dataset    Dataset for validation ( 'Thai_normal_alldpi', 'Thai_normal_200dpi', 'Thai_normal_400dpi', 'Thai_bold_alldpi', 'All_Thai_styles',
                 'Thai_English_normal', 'Thai_bold_400dpi', 'Thai_bold_italic', 'All_Thai_English_styles')
  test_dataset   Dataset for testing ( 'Thai_normal_alldpi', 'Thai_normal_200dpi', 'Thai_normal_400dpi', 'Thai_bold_alldpi', 'All_Thai_styles',
                 'Thai_English_normal', 'Thai_bold_400dpi', 'Thai_bold_italic', 'All_Thai_English_styles')
  directory      The path of the dataset (e.g., /scratch/lt2326-2926-h24/ThaiOCR/ThaiOCR-TrainigSet).

options:
  -h, --help     show this help message and exit
```

2. Creating the model & challenges

In order to get the training you have to run on the server this command :

```
python train.py --epochs 10 --batch_size 32
```

and for the test:

```
python test.py --epochs 10 --batch_size 32
```

model.py
I chose in my model to have three layer CNN with increasing filter sizes in order to capture more complex features of the images and I used MaxPooling after each layer so that the spatial dimensions can be reduced. I also used x.view(x.size(0), -1) to flatten the tensor for the fully connected layers, so I can get from the multidimensional convolutional layers, a 1-dimensional vector. I added a Dropout layer (0.5) to randomly drop neurons during training, helping the model generalize better and batch normalization to make training more stable.

dataset.py
In this script I have the extract_unique_labels() function in order to extract and sort the unique labels from the image paths. Also, inside the ThaiEngOCRDataset Class only the .bmp paths are stored and mapped to indices and returns the transformed image and its corresponding label.

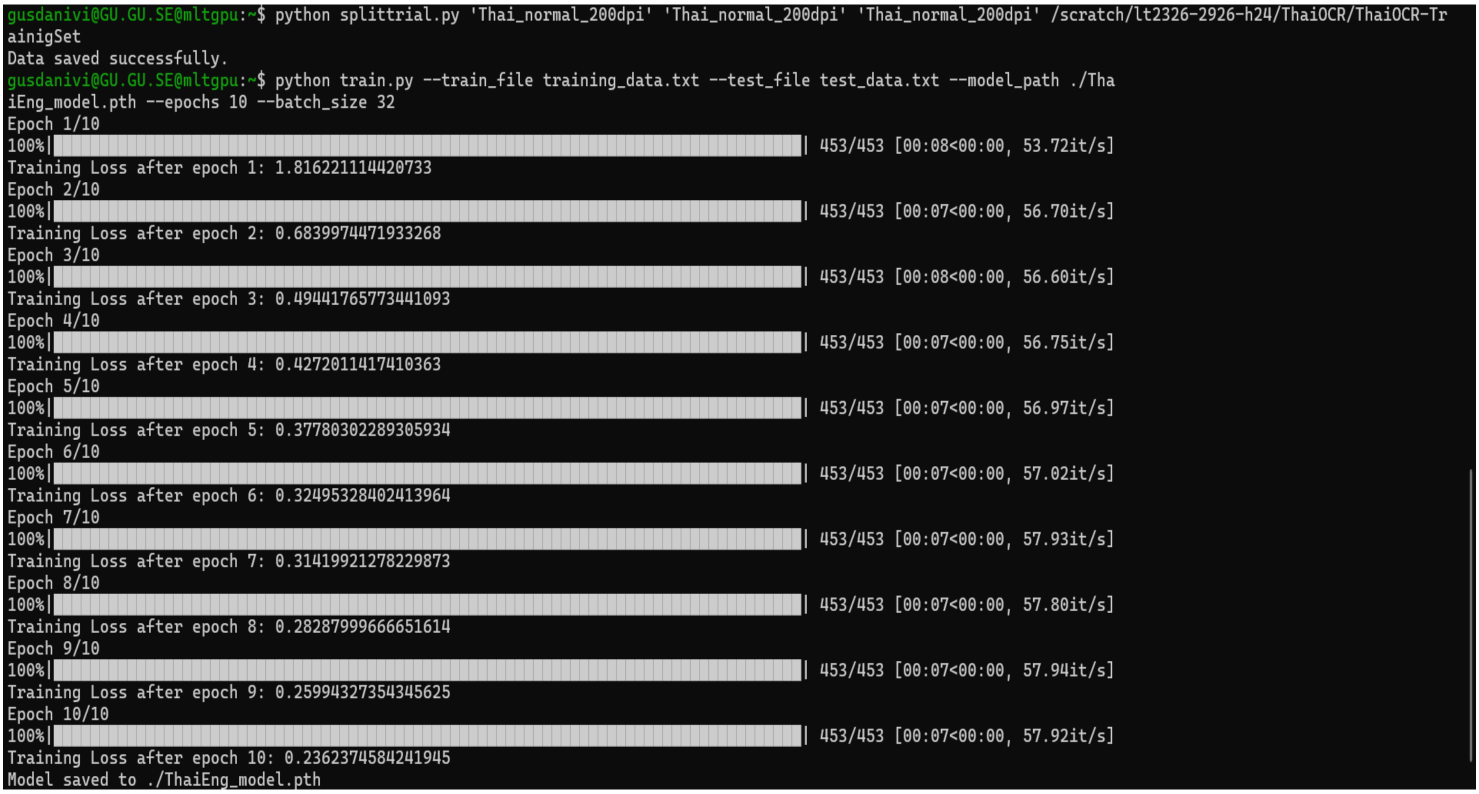
train.py
Training the model: In order to ensure that the images have the same format, I used transforms in order to resize the images to 32x32. I created a mapping from labels to indices and NLLLoss as a loss function. Before each batch I used optimizer.zero_grad() to clear gradients and I compute loss and update the model using backpropagation.

test.py
Evaluates the model using the sklearn metrics accuracy, precision, recall and F1-score on unseen data.

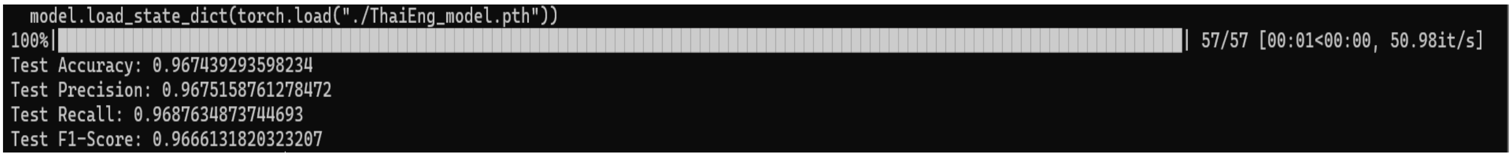
validation.py
Evaluates the Thai OCR model using the sklearn metrics accuracy, precision, recall and F1-score during training.

Checking all the possible character resolutions from the board by running the training we can observe that the loss is getting lower so the model is learning and improving its performance. Also the accuracy, precision, recall and F1-score are high which indicated that the model performs well. Below you will find screenshots with the training and testing from all the experiments I did with the training and testing data.

A. Training: Thai normal 200dpi, Testing: Thai normal 200dpi

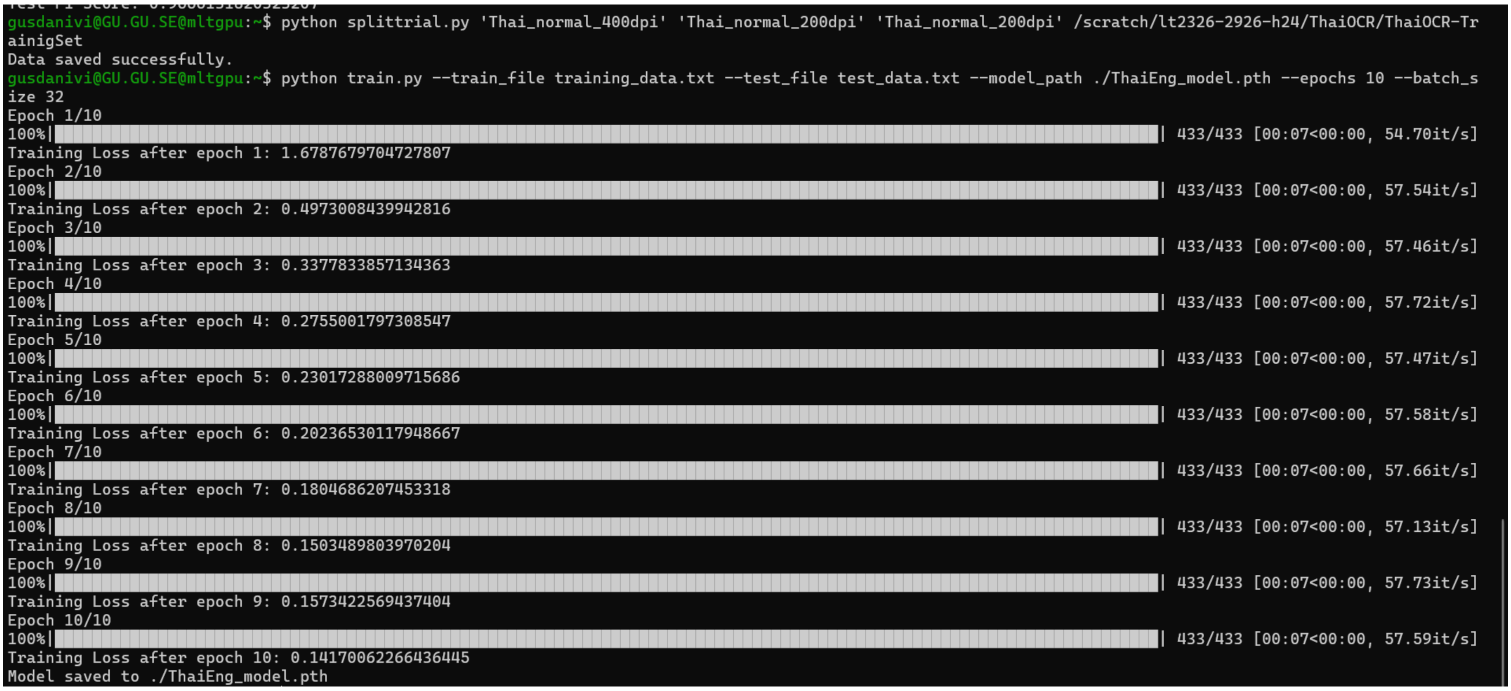


```
gusdanivi@GU.GU.SE@mltgpu:~$ python splittrial.py 'Thai_normal_200dpi' 'Thai_normal_200dpi' 'Thai_normal_200dpi' /scratch/lt2326-2926-h24/ThaiOCR/ThaiOCR-Tr
ainigSet
Data saved successfully.
gusdanivi@GU.GU.SE@mltgpu:~$ python train.py --train_file training_data.txt --test_file test_data.txt --model_path ./Tha
iEng_model.pth --epochs 10 --batch_size 32
Epoch 1/10
100%|████████████████████████████████████████████████████████| 453/453 [00:08<00:00, 53.72it/s]
Training Loss after epoch 1: 1.816221114420733
Epoch 2/10
100%|████████████████████████████████████████████████████████| 453/453 [00:07<00:00, 56.70it/s]
Training Loss after epoch 2: 0.6839974471933268
Epoch 3/10
100%|████████████████████████████████████████████████████████| 453/453 [00:08<00:00, 56.60it/s]
Training Loss after epoch 3: 0.49441765773441093
Epoch 4/10
100%|████████████████████████████████████████████████████████| 453/453 [00:07<00:00, 56.75it/s]
Training Loss after epoch 4: 0.4272011417410363
Epoch 5/10
100%|████████████████████████████████████████████████████████| 453/453 [00:07<00:00, 56.97it/s]
Training Loss after epoch 5: 0.37780302289305934
Epoch 6/10
100%|████████████████████████████████████████████████████████| 453/453 [00:07<00:00, 57.02it/s]
Training Loss after epoch 6: 0.32495328402413964
Epoch 7/10
100%|████████████████████████████████████████████████████████| 453/453 [00:07<00:00, 57.93it/s]
Training Loss after epoch 7: 0.31419921278229873
Epoch 8/10
100%|████████████████████████████████████████████████████████| 453/453 [00:07<00:00, 57.80it/s]
Training Loss after epoch 8: 0.28287999666651614
Epoch 9/10
100%|████████████████████████████████████████████████████████| 453/453 [00:07<00:00, 57.94it/s]
Training Loss after epoch 9: 0.25994327354345625
Epoch 10/10
100%|████████████████████████████████████████████████████████| 453/453 [00:07<00:00, 57.92it/s]
Training Loss after epoch 10: 0.23623745584241945
Model saved to ./ThaiEng_model.pth
```

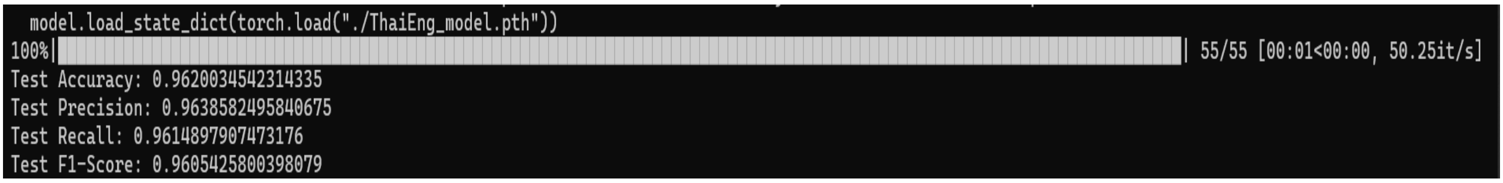


```
  model.load_state_dict(torch.load("./ThaiEng_model.pth"))
100%|████████████████████████████████████████████████████████████████████████| 57/57 [00:01<00:00, 50.98it/s]
Test Accuracy: 0.967439293598234
Test Precision: 0.9675158761278472
Test Recall: 0.9687634873744693
Test F1-Score: 0.9666131820323207
```

B. Training: Thai normal 200dpi , Testing: Thai normal 400 dpi

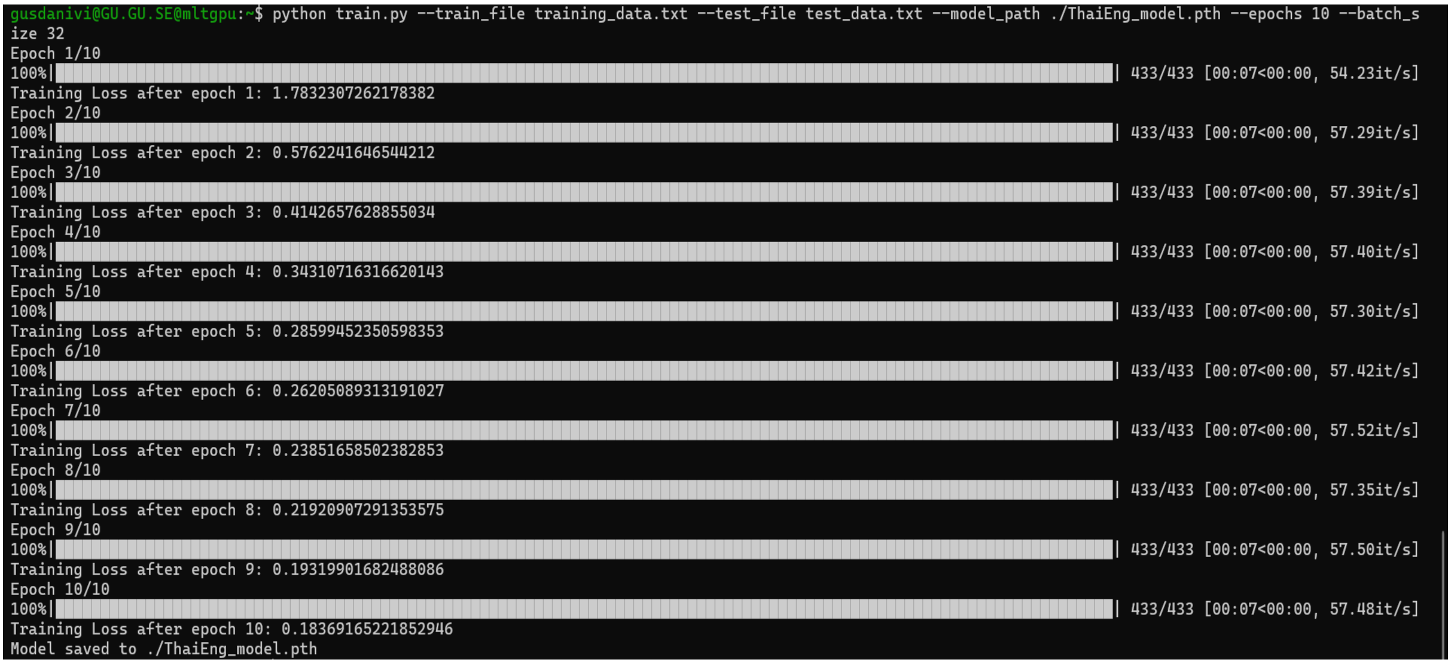


```
Test F1-Score: 0.9666131820323207
gusdanivi@GU.GU.SE@mltgpu:~$ python splittrial.py 'Thai_normal_400dpi' 'Thai_normal_200dpi' 'Thai_normal_200dpi' /scratch/lt2326-2926-h24/ThaiOCR/ThaiOCR-Tr
ainigSet
Data saved successfully.
gusdanivi@GU.GU.SE@mltgpu:~$ python train.py --train_file training_data.txt --test_file test_data.txt --model_path ./ThaiEng_model.pth --epochs 10 --batch_s
ize 32
Epoch 1/10
100%|██████████████████████████████████████████████████████████████████████████| 433/433 [00:07<00:00, 54.70it/s]
Training Loss after epoch 1: 1.67876797704727807
Epoch 2/10
100%|██████████████████████████████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.54it/s]
Training Loss after epoch 2: 0.4973008439942816
Epoch 3/10
100%|██████████████████████████████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.46it/s]
Training Loss after epoch 3: 0.3377833857134363
Epoch 4/10
100%|██████████████████████████████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.72it/s]
Training Loss after epoch 4: 0.2755001797308547
Epoch 5/10
100%|██████████████████████████████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.47it/s]
Training Loss after epoch 5: 0.23017288009715686
Epoch 6/10
100%|██████████████████████████████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.58it/s]
Training Loss after epoch 6: 0.20236530117948667
Epoch 7/10
100%|██████████████████████████████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.66it/s]
Training Loss after epoch 7: 0.1804686207453318
Epoch 8/10
100%|██████████████████████████████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.13it/s]
Training Loss after epoch 8: 0.1503489803970204
Epoch 9/10
100%|██████████████████████████████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.73it/s]
Training Loss after epoch 9: 0.1573422569437404
Epoch 10/10
100%|██████████████████████████████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.59it/s]
Training Loss after epoch 10: 0.14170062266436445
Model saved to ./ThaiEng_model.pth
```

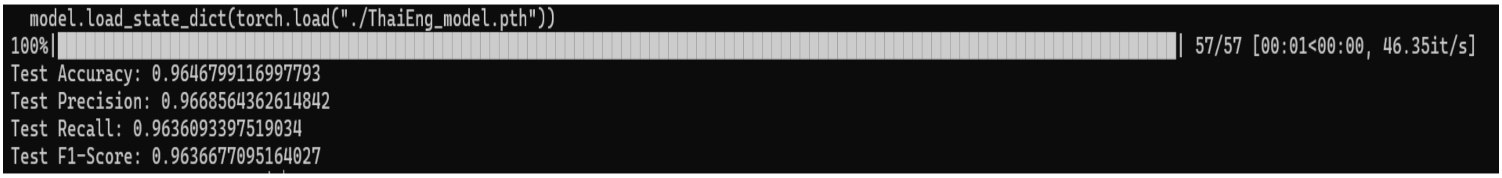


```
  model.load_state_dict(torch.load("./ThaiEng_model.pth"))
100%|██████████████████████████████████████████████████| 55/55 [00:01<00:00, 50.25it/s]
Test Accuracy: 0.9620034542314335
Test Precision: 0.9638582495840675
Test Recall: 0.9614897907473176
Test F1-Score: 0.9605425800398079
```

C. Training: Thai normal 400 dpi , Testing: Thai bold 400 dpi

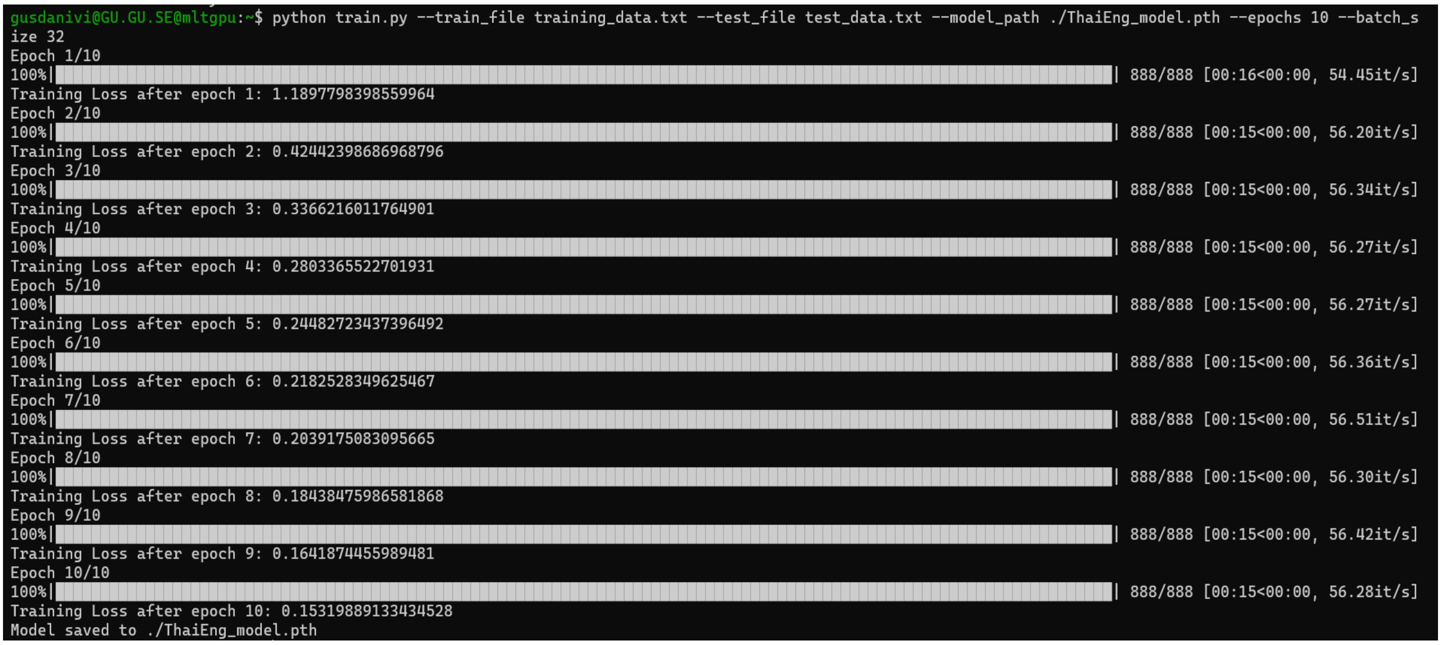


```
gusdanivi@GU.GU.SE@mltgpu:~$ python train.py --train_file training_data.txt --test_file test_data.txt --model_path ./ThaiEng_model.pth --epochs 10 --batch_s
ize 32
Epoch 1/10
100%|██████████████████████████████████████████████████| 433/433 [00:07<00:00, 54.23it/s]
Training Loss after epoch 1: 1.7832307262178382
Epoch 2/10
100%|██████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.29it/s]
Training Loss after epoch 2: 0.5762241646544212
Epoch 3/10
100%|██████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.39it/s]
Training Loss after epoch 3: 0.4142657628855034
Epoch 4/10
100%|██████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.40it/s]
Training Loss after epoch 4: 0.34310716316620143
Epoch 5/10
100%|██████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.30it/s]
Training Loss after epoch 5: 0.28599452350598353
Epoch 6/10
100%|██████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.42it/s]
Training Loss after epoch 6: 0.26205089313191027
Epoch 7/10
100%|██████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.52it/s]
Training Loss after epoch 7: 0.23851658502382853
Epoch 8/10
100%|██████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.35it/s]
Training Loss after epoch 8: 0.21920907291353575
Epoch 9/10
100%|██████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.50it/s]
Training Loss after epoch 9: 0.19319901682488086
Epoch 10/10
100%|██████████████████████████████████████████████████| 433/433 [00:07<00:00, 57.48it/s]
Training Loss after epoch 10: 0.18369165221852946
Model saved to ./ThaiEng_model.pth
```

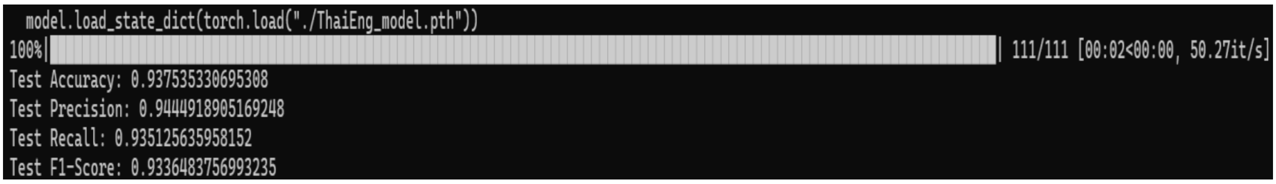


```
  model.load_state_dict(torch.load("./ThaiEng_model.pth"))
100%|██████████████████████████████████████████████████| 57/57 [00:01<00:00, 46.35it/s]
Test Accuracy: 0.9646799116997793
Test Precision: 0.9668564362614842
Test Recall: 0.9636093397519034
Test F1-Score: 0.9636677095164027
```

D. Training: Thai bold all dpi ,Testing: Thai normal all dpi

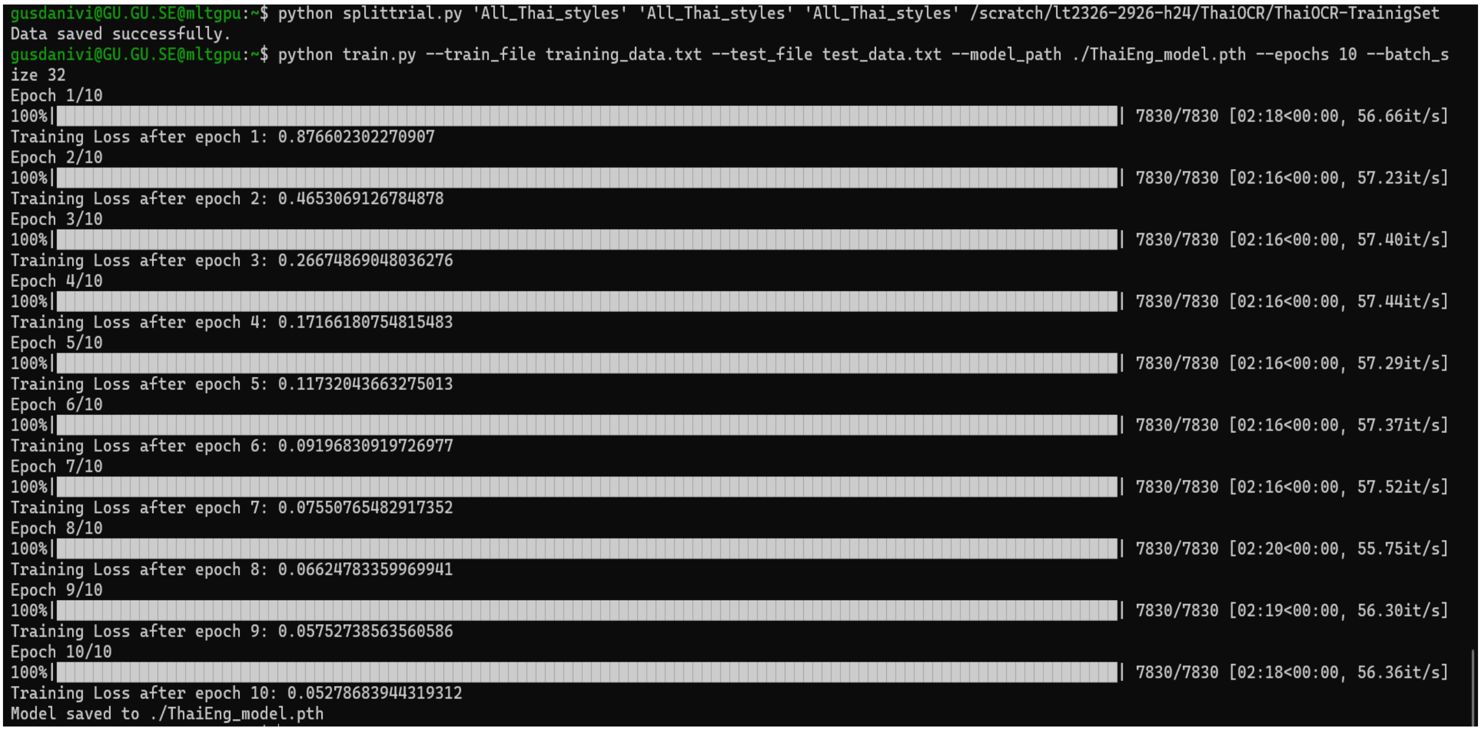


```
gusdanivi@GU.GU.SE@mltgpu:~$ python train.py --train_file training_data.txt --test_file test_data.txt --model_path ./ThaiEng_model.pth --epochs 10 --batch_s
ize 32
Epoch 1/10
100%|██████████████████████████████████████████████████| 888/888 [00:16<00:00, 54.45it/s]
Training Loss after epoch 1: 1.1897798398559964
Epoch 2/10
100%|██████████████████████████████████████████████████| 888/888 [00:15<00:00, 56.20it/s]
Training Loss after epoch 2: 0.42442398686968796
Epoch 3/10
100%|██████████████████████████████████████████████████| 888/888 [00:15<00:00, 56.34it/s]
Training Loss after epoch 3: 0.3366216011764901
Epoch 4/10
100%|██████████████████████████████████████████████████| 888/888 [00:15<00:00, 56.27it/s]
Training Loss after epoch 4: 0.2803365522701931
Epoch 5/10
100%|██████████████████████████████████████████████████| 888/888 [00:15<00:00, 56.27it/s]
Training Loss after epoch 5: 0.24482723437396492
Epoch 6/10
100%|██████████████████████████████████████████████████| 888/888 [00:15<00:00, 56.36it/s]
Training Loss after epoch 6: 0.2182528349625467
Epoch 7/10
100%|██████████████████████████████████████████████████| 888/888 [00:15<00:00, 56.51it/s]
Training Loss after epoch 7: 0.2039175083095665
Epoch 8/10
100%|██████████████████████████████████████████████████| 888/888 [00:15<00:00, 56.30it/s]
Training Loss after epoch 8: 0.18438475986581868
Epoch 9/10
100%|██████████████████████████████████████████████████| 888/888 [00:15<00:00, 56.42it/s]
Training Loss after epoch 9: 0.1641874455989481
Epoch 10/10
100%|██████████████████████████████████████████████████| 888/888 [00:15<00:00, 56.28it/s]
Training Loss after epoch 10: 0.15319889133434528
Model saved to ./ThaiEng_model.pth
```

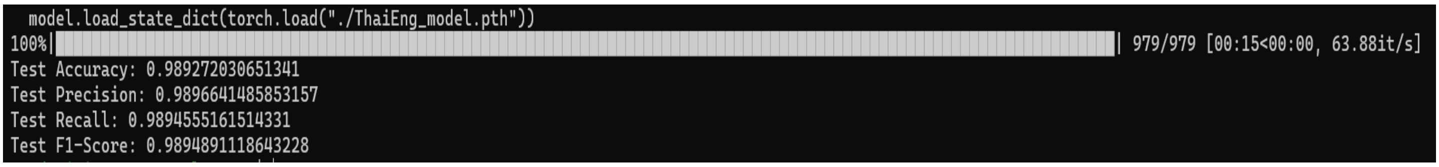


```
  model.load_state_dict(torch.load("./ThaiEng_model.pth"))
100%|██████████████████████████████████████████████████████████████████| 111/111 [00:02<00:00, 50.27it/s]
Test Accuracy: 0.937535330695308
Test Precision: 0.9444918905169248
Test Recall: 0.935125635958152
Test F1-Score: 0.9336483756993235
```

E. Training & Testing : All Thai styles

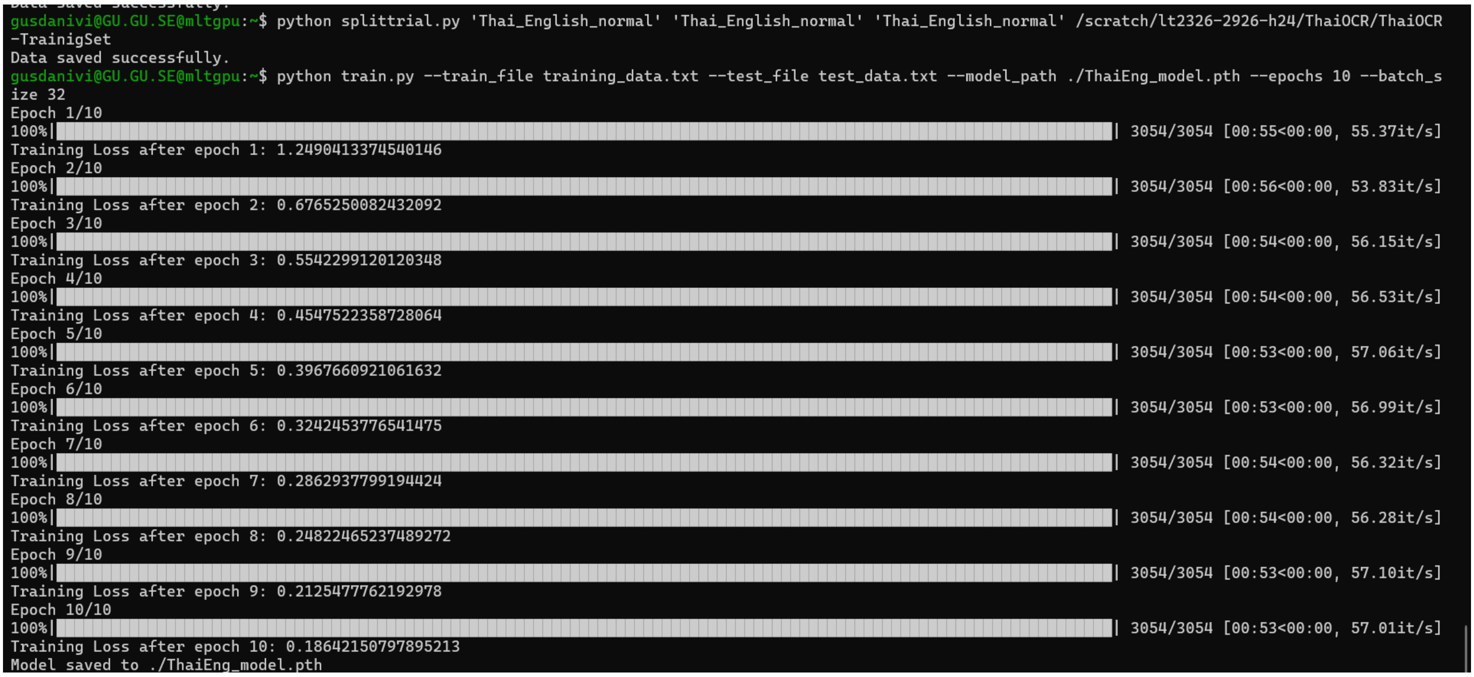


```
gusdanivi@GU.GU.SE@mltgpu:~$ python splittrial.py 'All_Thai_styles' 'All_Thai_styles' 'All_Thai_styles' /scratch/lt2326-2926-h24/ThaiOCR/ThaiOCR-TrainigSet
Data saved successfully.
gusdanivi@GU.GU.SE@mltgpu:~$ python train.py --train_file training_data.txt --test_file test_data.txt --model_path ./ThaiEng_model.pth --epochs 10 --batch_s
ize 32
Epoch 1/10
100%|██████████████████████████████████████████████████████████████████| 7830/7830 [02:18<00:00, 56.66it/s]
Training Loss after epoch 1: 0.876602302270907
Epoch 2/10
100%|██████████████████████████████████████████████████████████████████| 7830/7830 [02:16<00:00, 57.23it/s]
Training Loss after epoch 2: 0.4653069126784878
Epoch 3/10
100%|██████████████████████████████████████████████████████████████████| 7830/7830 [02:16<00:00, 57.40it/s]
Training Loss after epoch 3: 0.26674869048036276
Epoch 4/10
100%|██████████████████████████████████████████████████████████████████| 7830/7830 [02:16<00:00, 57.44it/s]
Training Loss after epoch 4: 0.17166180754815483
Epoch 5/10
100%|██████████████████████████████████████████████████████████████████| 7830/7830 [02:16<00:00, 57.29it/s]
Training Loss after epoch 5: 0.11732043663275013
Epoch 6/10
100%|██████████████████████████████████████████████████████████████████| 7830/7830 [02:16<00:00, 57.37it/s]
Training Loss after epoch 6: 0.09196830919726977
Epoch 7/10
100%|██████████████████████████████████████████████████████████████████| 7830/7830 [02:16<00:00, 57.52it/s]
Training Loss after epoch 7: 0.07550765482917352
Epoch 8/10
100%|██████████████████████████████████████████████████████████████████| 7830/7830 [02:20<00:00, 55.75it/s]
Training Loss after epoch 8: 0.06624783359969941
Epoch 9/10
100%|██████████████████████████████████████████████████████████████████| 7830/7830 [02:19<00:00, 56.30it/s]
Training Loss after epoch 9: 0.05752738563560586
Epoch 10/10
100%|██████████████████████████████████████████████████████████████████| 7830/7830 [02:18<00:00, 56.36it/s]
Training Loss after epoch 10: 0.05278683944319312
Model saved to ./ThaiEng_model.pth
```

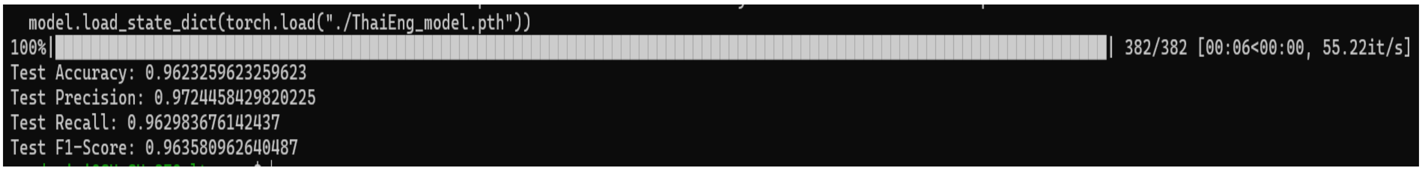


```
  model.load_state_dict(torch.load("./ThaiEng_model.pth"))
100%|██████████████████████████████████████████████████████████████████| 979/979 [00:15<00:00, 63.88it/s]
Test Accuracy: 0.989272030651341
Test Precision: 0.9896641485853157
Test Recall: 0.9894555161514331
Test F1-Score: 0.9894891118643228
```

F. Training & Testing : Thai and English normal

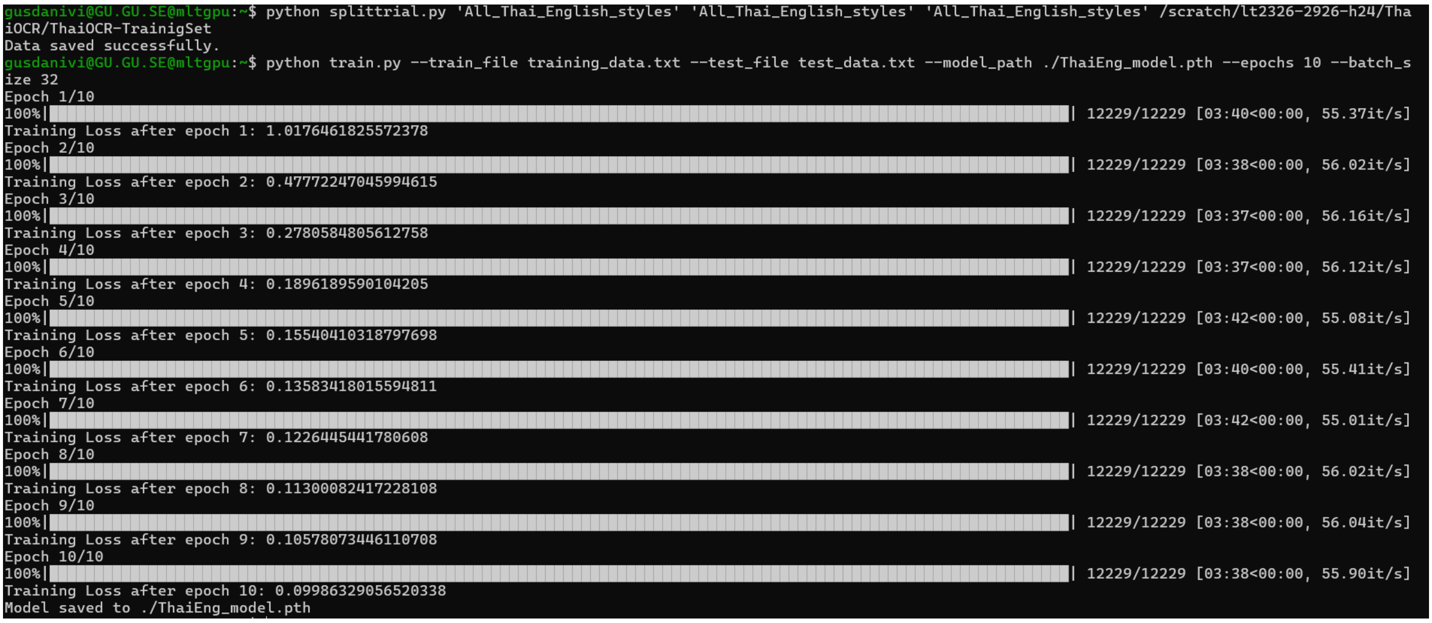


```
Data saved successfully.
gusdanivi@GU.GU.SE@mltgpu:~$ python splittrial.py 'Thai_English_normal' 'Thai_English_normal' 'Thai_English_normal' /scratch/lt2326-2926-h24/ThaiOCR/ThaiOCR
-TrainigSet
Data saved successfully.
gusdanivi@GU.GU.SE@mltgpu:~$ python train.py --train_file training_data.txt --test_file test_data.txt --model_path ./ThaiEng_model.pth --epochs 10 --batch_s
ize 32
Epoch 1/10
100%|████████████████████████████████████████████████████████████| 3054/3054 [00:55<00:00, 55.37it/s]
Training Loss after epoch 1: 1.2490413374540146
Epoch 2/10
100%|████████████████████████████████████████████████████████████| 3054/3054 [00:56<00:00, 53.83it/s]
Training Loss after epoch 2: 0.6765250082432092
Epoch 3/10
100%|████████████████████████████████████████████████████████████| 3054/3054 [00:54<00:00, 56.15it/s]
Training Loss after epoch 3: 0.5542299120120348
Epoch 4/10
100%|████████████████████████████████████████████████████████████| 3054/3054 [00:54<00:00, 56.53it/s]
Training Loss after epoch 4: 0.4547522358728064
Epoch 5/10
100%|████████████████████████████████████████████████████████████| 3054/3054 [00:53<00:00, 57.06it/s]
Training Loss after epoch 5: 0.3967660921061632
Epoch 6/10
100%|████████████████████████████████████████████████████████████| 3054/3054 [00:53<00:00, 56.99it/s]
Training Loss after epoch 6: 0.3242453776541475
Epoch 7/10
100%|████████████████████████████████████████████████████████████| 3054/3054 [00:54<00:00, 56.32it/s]
Training Loss after epoch 7: 0.2862937799194424
Epoch 8/10
100%|████████████████████████████████████████████████████████████| 3054/3054 [00:54<00:00, 56.28it/s]
Training Loss after epoch 8: 0.24822465237489272
Epoch 9/10
100%|████████████████████████████████████████████████████████████| 3054/3054 [00:53<00:00, 57.10it/s]
Training Loss after epoch 9: 0.2125477762192978
Epoch 10/10
100%|████████████████████████████████████████████████████████████| 3054/3054 [00:53<00:00, 57.01it/s]
Training Loss after epoch 10: 0.18642150797895213
Model saved to ./ThaiEng_model.pth
```

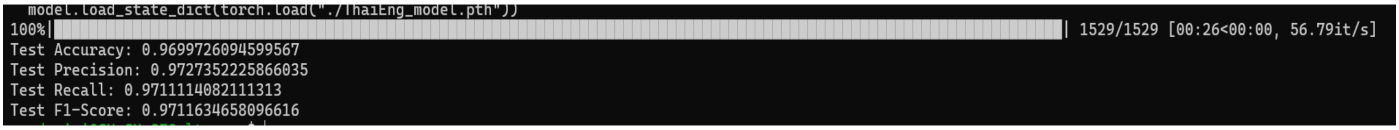


```
  model.load_state_dict(torch.load("./ThaiEng_model.pth"))
100%|████████████████████████████████████████████████████████████| 382/382 [00:06<00:00, 55.22it/s]
Test Accuracy: 0.9623259623259623
Test Precision: 0.9724458429820225
Test Recall: 0.962983676142437
Test F1-Score: 0.963580962640487
```

G. Training & Testing : Thai & English all styles



```
gusdanivi@GU.GU.SE@mltgpu:~$ python splittrial.py 'All_Thai_English_styles' 'All_Thai_English_styles' 'All_Thai_English_styles' /scratch/lt2326-2926-h24/Tha
iOCR/ThaiOCR-TrainigSet
Data saved successfully.
gusdanivi@GU.GU.SE@mltgpu:~$ python train.py --train_file training_data.txt --test_file test_data.txt --model_path ./ThaiEng_model.pth --epochs 10 --batch_s
ize 32
Epoch 1/10
100%|████████████████████████████████████████████████████████████| 12229/12229 [03:40<00:00, 55.37it/s]
Training Loss after epoch 1: 1.0176461825572378
Epoch 2/10
100%|████████████████████████████████████████████████████████████| 12229/12229 [03:38<00:00, 56.02it/s]
Training Loss after epoch 2: 0.47772247045994615
Epoch 3/10
100%|████████████████████████████████████████████████████████████| 12229/12229 [03:37<00:00, 56.16it/s]
Training Loss after epoch 3: 0.2780584805612758
Epoch 4/10
100%|████████████████████████████████████████████████████████████| 12229/12229 [03:37<00:00, 56.12it/s]
Training Loss after epoch 4: 0.1896189590104205
Epoch 5/10
100%|████████████████████████████████████████████████████████████| 12229/12229 [03:42<00:00, 55.08it/s]
Training Loss after epoch 5: 0.15540410318797698
Epoch 6/10
100%|████████████████████████████████████████████████████████████| 12229/12229 [03:40<00:00, 55.41it/s]
Training Loss after epoch 6: 0.13583418015594811
Epoch 7/10
100%|████████████████████████████████████████████████████████████| 12229/12229 [03:42<00:00, 55.01it/s]
Training Loss after epoch 7: 0.1226445441780608
Epoch 8/10
100%|████████████████████████████████████████████████████████████| 12229/12229 [03:38<00:00, 56.02it/s]
Training Loss after epoch 8: 0.11300082417228108
Epoch 9/10
100%|████████████████████████████████████████████████████████████| 12229/12229 [03:38<00:00, 56.04it/s]
Training Loss after epoch 9: 0.10578073446110708
Epoch 10/10
100%|████████████████████████████████████████████████████████████| 12229/12229 [03:38<00:00, 55.90it/s]
Training Loss after epoch 10: 0.09986329056520338
Model saved to ./ThaiEng_model.pth
```



```
  model.load_state_dict(torch.load("./ThaiEng_model.pth"))
100%|████████████████████████████████████████████████████████████| 1529/1529 [00:26<00:00, 56.79it/s]
Test Accuracy: 0.9699726094599567
Test Precision: 0.9727352225866035
Test Recall: 0.9711114082111313
Test F1-Score: 0.9711634658096616
```