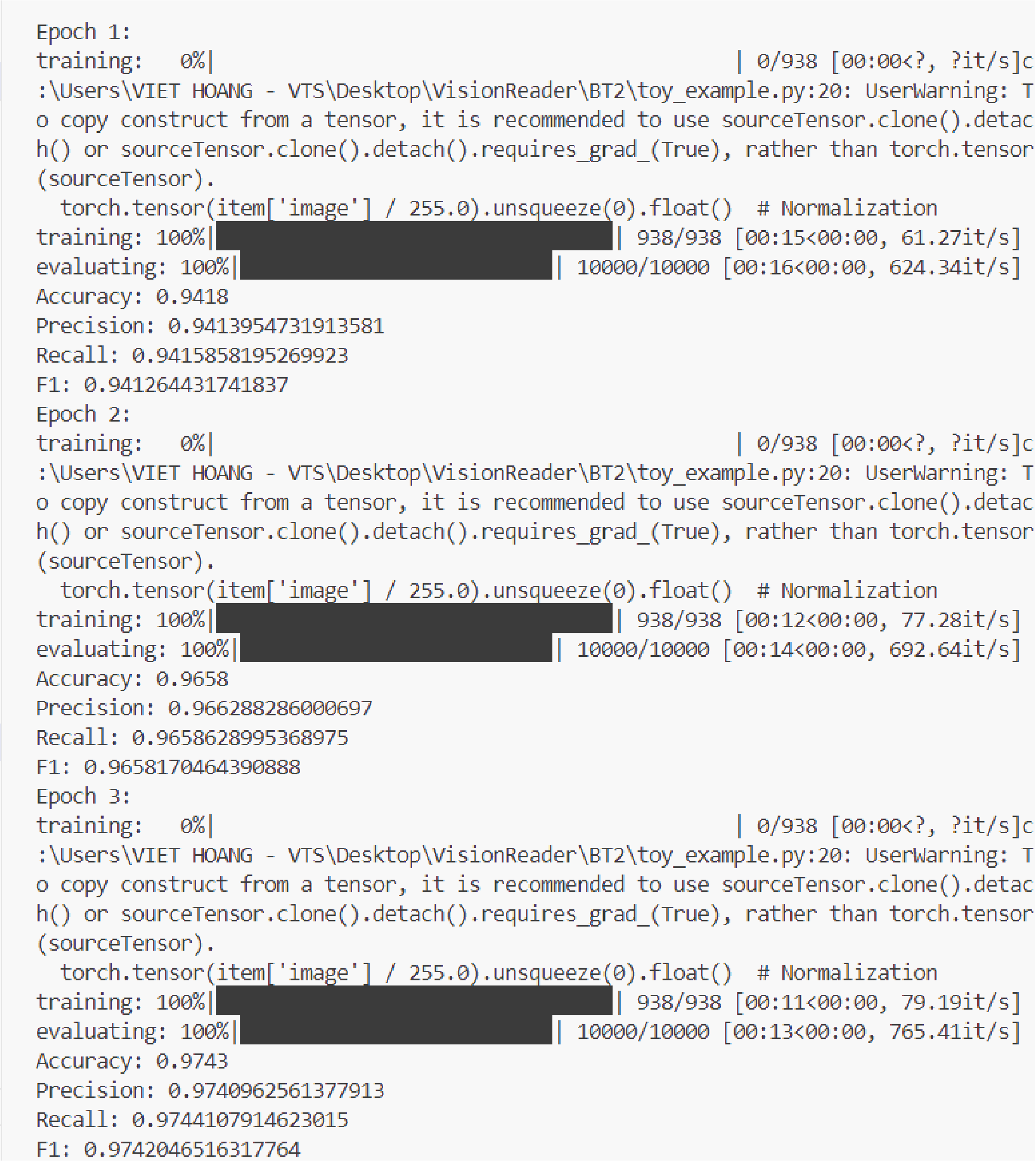
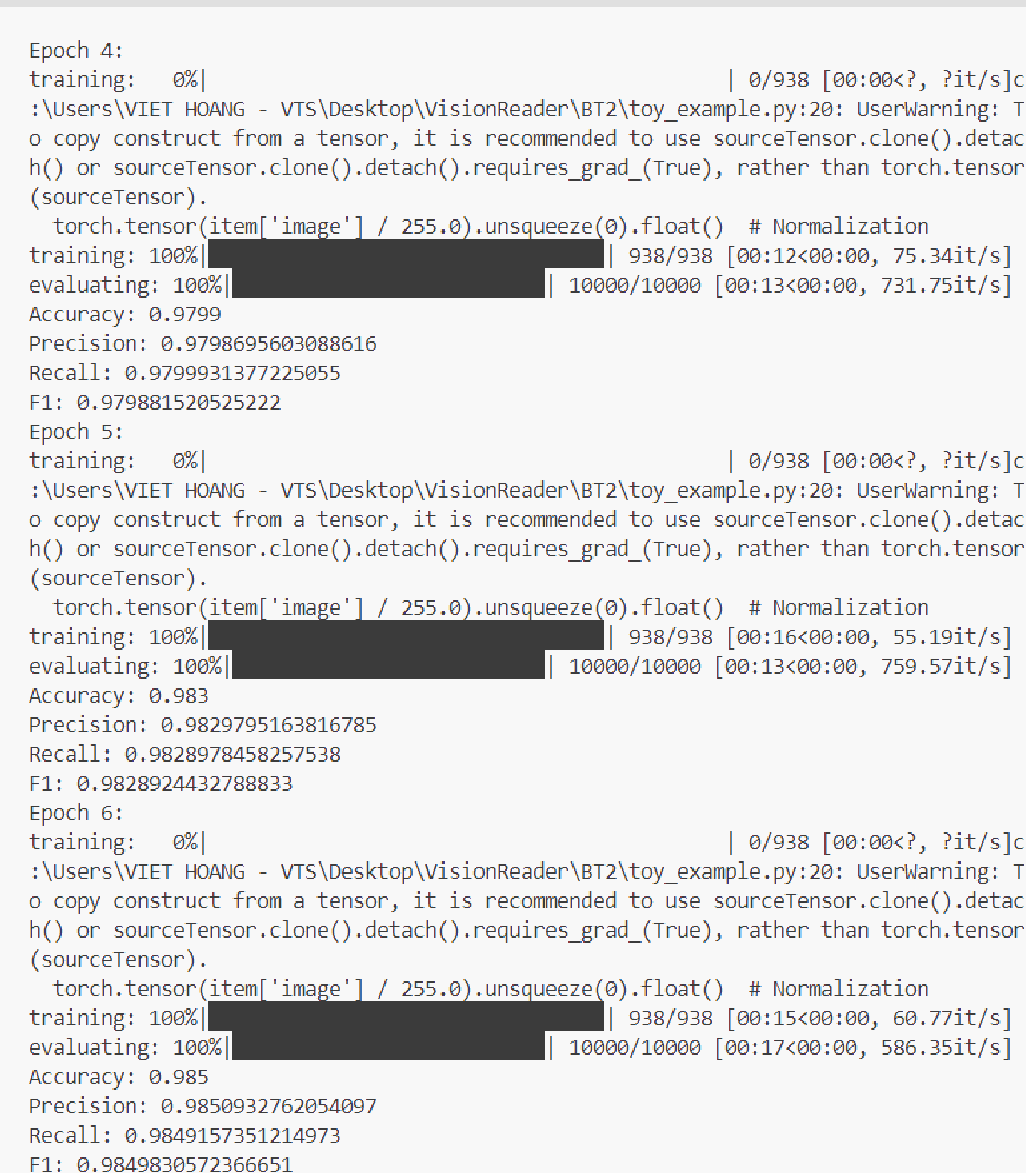
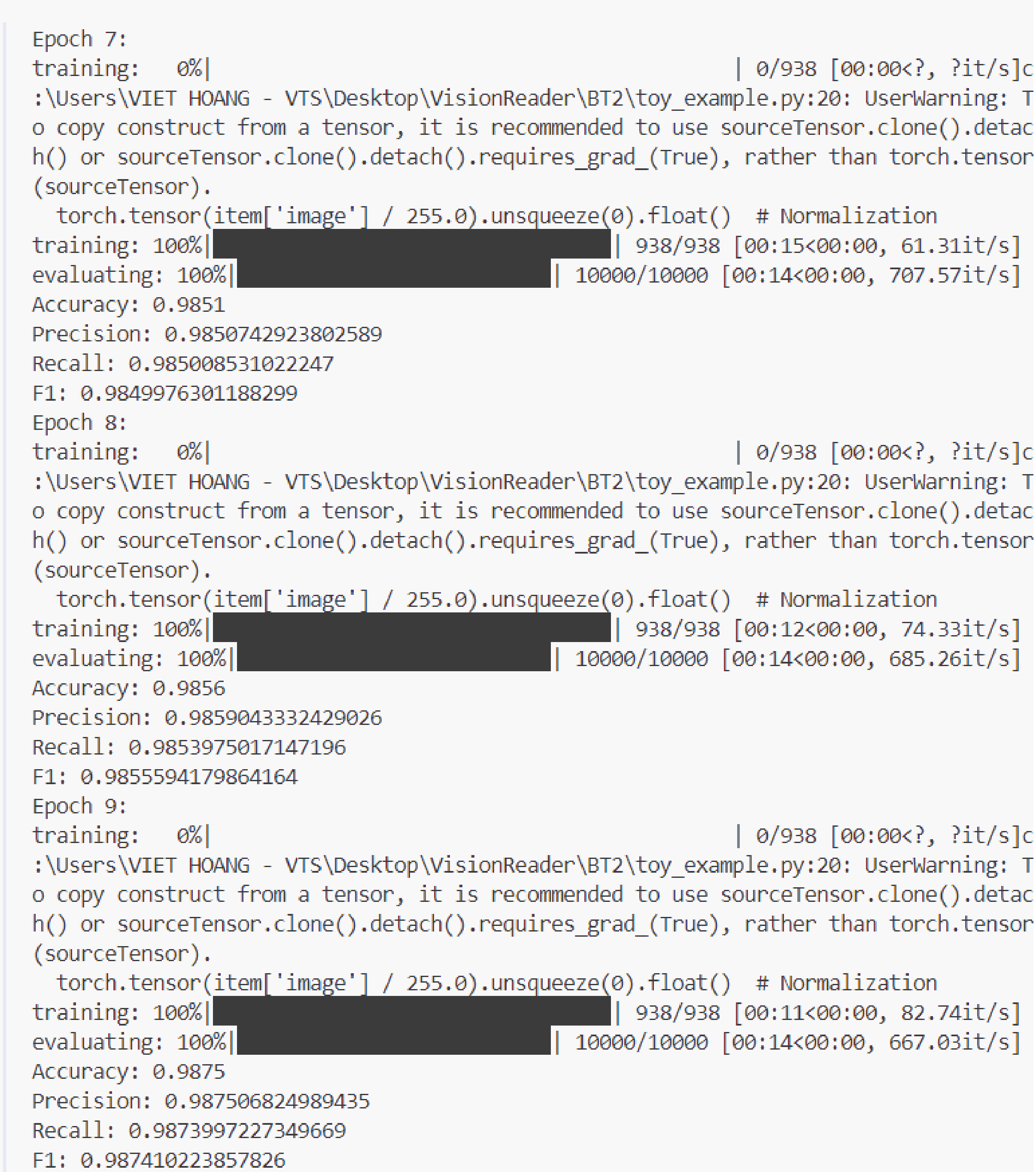
Câu 1:



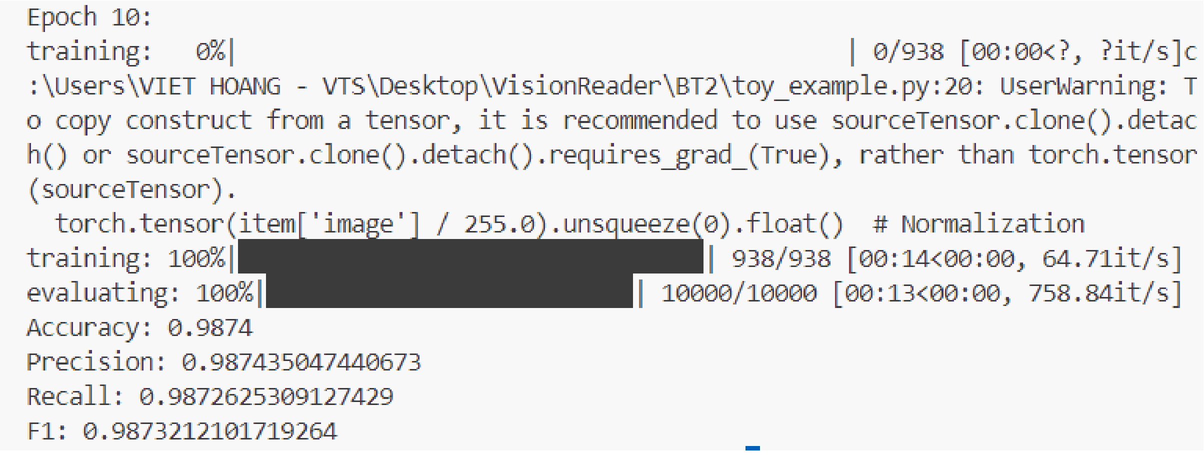
```
Epoch 1:
training:   0%|                                          | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|████████████████████████████████| 938/938 [00:15<00:00, 61.27it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [00:16<00:00, 624.34it/s]
Accuracy: 0.9418
Precision: 0.9413954731913581
Recall: 0.9415858195269923
F1: 0.941264431741837
Epoch 2:
training:   0%|                                          | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|████████████████████████████████| 938/938 [00:12<00:00, 77.28it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [00:14<00:00, 692.64it/s]
Accuracy: 0.9658
Precision: 0.966288286000697
Recall: 0.9658628995368975
F1: 0.9658170464390888
Epoch 3:
training:   0%|                                          | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|████████████████████████████████| 938/938 [00:11<00:00, 79.19it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [00:13<00:00, 765.41it/s]
Accuracy: 0.9743
Precision: 0.9740962561377913
Recall: 0.9744107914623015
F1: 0.9742046516317764
```



```
Epoch 4:
training:   0%|                                          | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|████████████████████████████████| 938/938 [00:12<00:00, 75.34it/s]
evaluating: 100%|█████████████████████████| 10000/10000 [00:13<00:00, 731.75it/s]
Accuracy: 0.9799
Precision: 0.9798695603088616
Recall: 0.9799931377225055
F1: 0.979881520525222
Epoch 5:
training:   0%|                                          | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|████████████████████████████████| 938/938 [00:16<00:00, 55.19it/s]
evaluating: 100%|█████████████████████████| 10000/10000 [00:13<00:00, 759.57it/s]
Accuracy: 0.983
Precision: 0.9829795163816785
Recall: 0.9828978458257538
F1: 0.9828924432788833
Epoch 6:
training:   0%|                                          | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|████████████████████████████████| 938/938 [00:15<00:00, 60.77it/s]
evaluating: 100%|█████████████████████████| 10000/10000 [00:17<00:00, 586.35it/s]
Accuracy: 0.985
Precision: 0.9850932762054097
Recall: 0.9849157351214973
F1: 0.9849830572366651
```



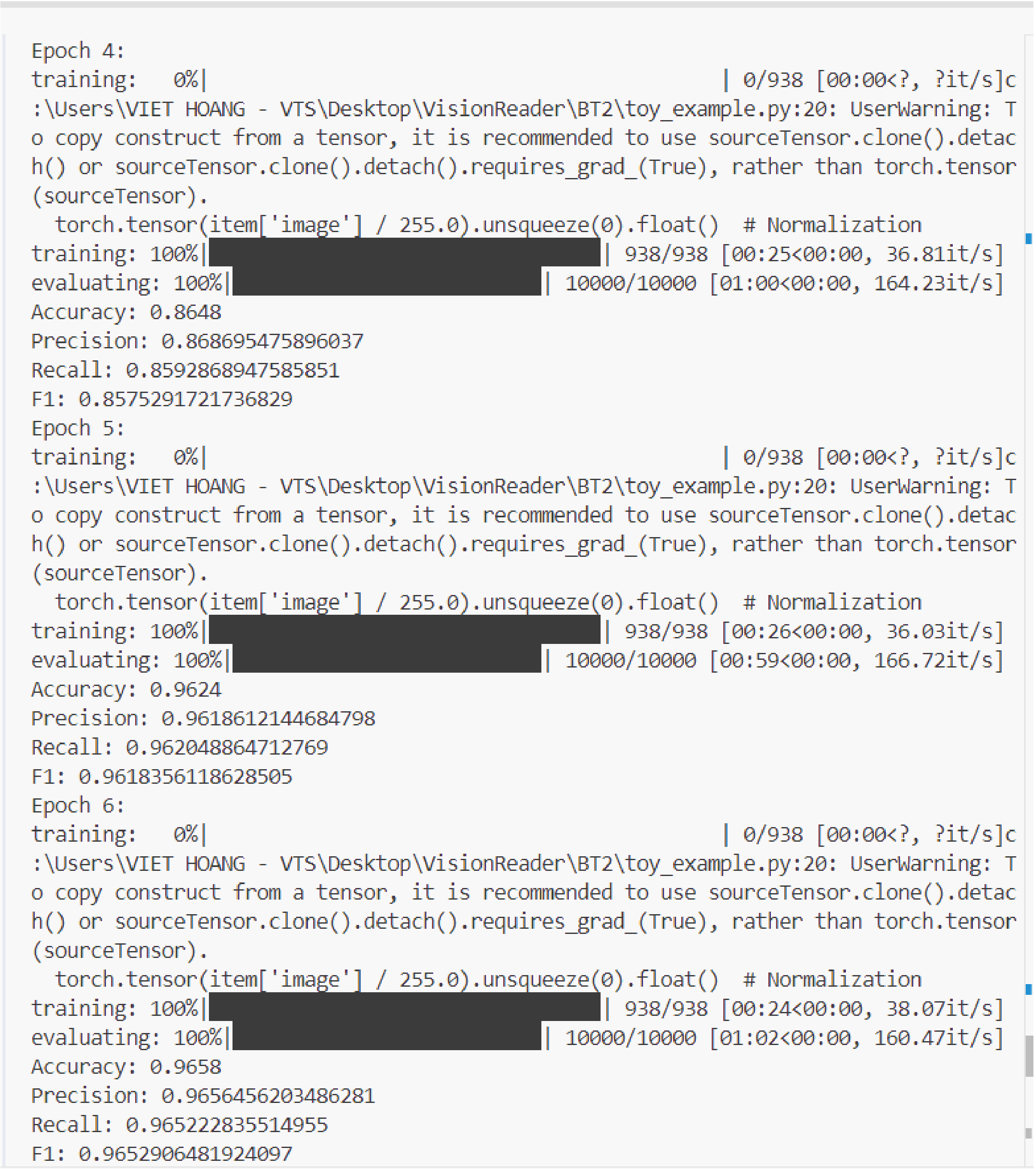
```
Epoch 7:
training:   0%|                                            | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [00:15<00:00, 61.31it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [00:14<00:00, 707.57it/s]
Accuracy: 0.9851
Precision: 0.9850742923802589
Recall: 0.985008531022247
F1: 0.9849976301188299
Epoch 8:
training:   0%|                                            | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [00:12<00:00, 74.33it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [00:14<00:00, 685.26it/s]
Accuracy: 0.9856
Precision: 0.9859043332429026
Recall: 0.9853975017147196
F1: 0.9855594179864164
Epoch 9:
training:   0%|                                            | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [00:11<00:00, 82.74it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [00:14<00:00, 667.03it/s]
Accuracy: 0.9875
Precision: 0.987506824989435
Recall: 0.9873997227349669
F1: 0.987410223857826
```



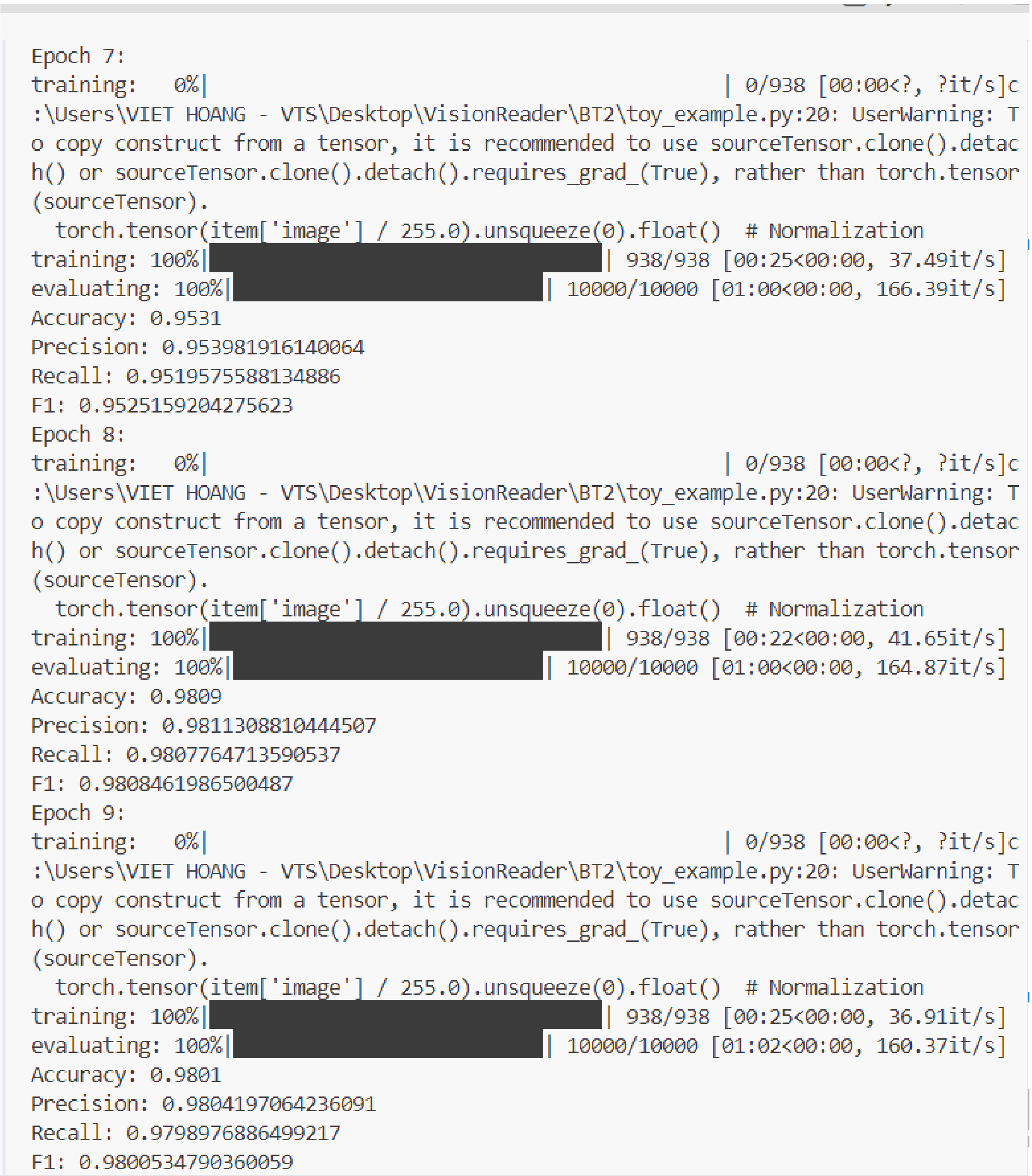
```
Epoch 10:
training:   0%|                                            | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [00:14<00:00, 64.71it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [00:13<00:00, 758.84it/s]
Accuracy: 0.9874
Precision: 0.987435047440673
Recall: 0.9872625309127429
F1: 0.9873212101719264
```

Câu 2

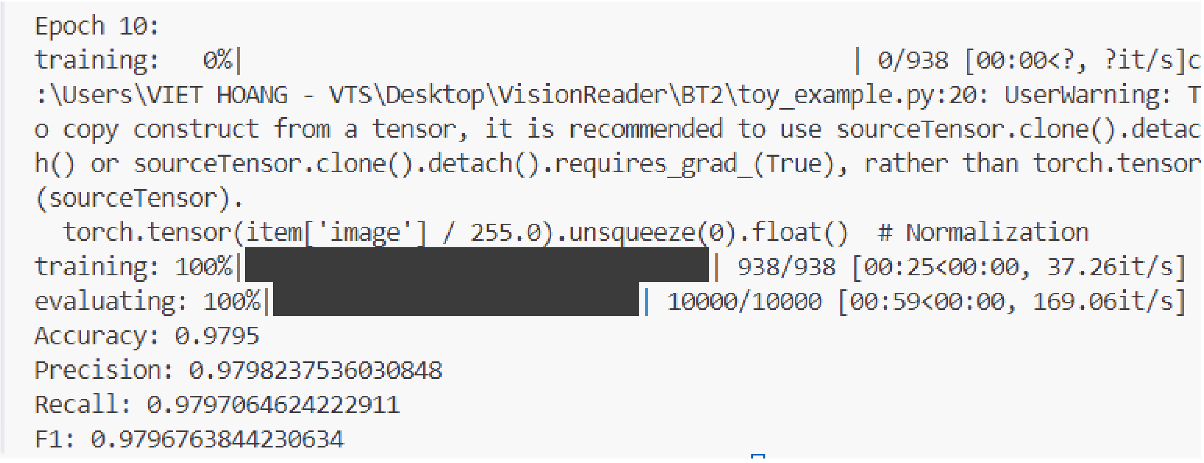
```
Epoch 1:
training:   0%|                                            | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [00:26<00:00, 35.43it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [00:58<00:00, 171.10it/s]
Accuracy: 0.6363
Precision: 0.624995856576714
Recall: 0.626524433478665
F1: 0.5943451157789676
Epoch 2:
training:   0%|                                            | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [00:42<00:00, 22.12it/s]
evaluating: 100%|███████████████████████████| 10000/10000 [02:51<00:00, 58.26it/s]
Accuracy: 0.7712
Precision: 0.7505088344007814
Recall: 0.7623525611135584
F1: 0.7476866285561685
Epoch 3:
training:   0%|                                            | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [00:41<00:00, 22.42it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [01:03<00:00, 158.11it/s]
Accuracy: 0.8301
Precision: 0.8316344414789384
Recall: 0.8258833504809855
F1: 0.8249862775485095
```



```
Epoch 4:
training:   0%|                                                  | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|██████████████████████████████| 938/938 [00:25<00:00, 36.81it/s]
evaluating: 100%|████████████████████████| 10000/10000 [01:00<00:00, 164.23it/s]
Accuracy: 0.8648
Precision: 0.868695475896037
Recall: 0.8592868947585851
F1: 0.8575291721736829
Epoch 5:
training:   0%|                                                  | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|██████████████████████████████| 938/938 [00:26<00:00, 36.03it/s]
evaluating: 100%|████████████████████████| 10000/10000 [00:59<00:00, 166.72it/s]
Accuracy: 0.9624
Precision: 0.9618612144684798
Recall: 0.962048864712769
F1: 0.9618356118628505
Epoch 6:
training:   0%|                                                  | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|██████████████████████████████| 938/938 [00:24<00:00, 38.07it/s]
evaluating: 100%|████████████████████████| 10000/10000 [01:02<00:00, 160.47it/s]
Accuracy: 0.9658
Precision: 0.9656456203486281
Recall: 0.965222835514955
F1: 0.9652906481924097
```

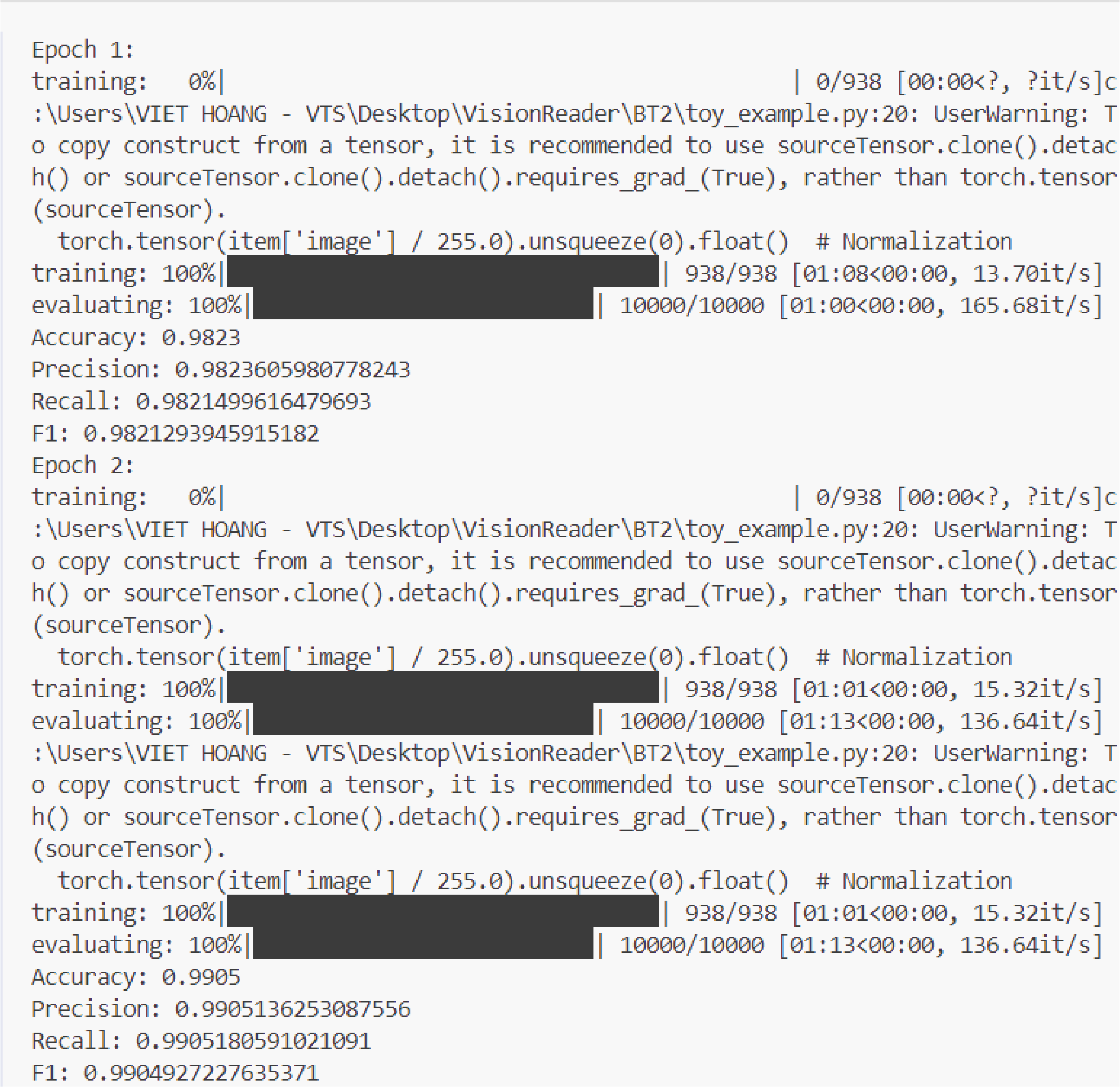


```
Epoch 7:
training:   0%|                                              | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|████████████████████████████████| 938/938 [00:25<00:00, 37.49it/s]
evaluating: 100%|█████████████████████████| 10000/10000 [01:00<00:00, 166.39it/s]
Accuracy: 0.9531
Precision: 0.953981916140064
Recall: 0.9519575588134886
F1: 0.9525159204275623
Epoch 8:
training:   0%|                                              | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|████████████████████████████████| 938/938 [00:22<00:00, 41.65it/s]
evaluating: 100%|█████████████████████████| 10000/10000 [01:00<00:00, 164.87it/s]
Accuracy: 0.9809
Precision: 0.9811308810444507
Recall: 0.9807764713590537
F1: 0.9808461986500487
Epoch 9:
training:   0%|                                              | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|████████████████████████████████| 938/938 [00:25<00:00, 36.91it/s]
evaluating: 100%|█████████████████████████| 10000/10000 [01:02<00:00, 160.37it/s]
Accuracy: 0.9801
Precision: 0.9804197064236091
Recall: 0.9798976886499217
F1: 0.9800534790360059
```

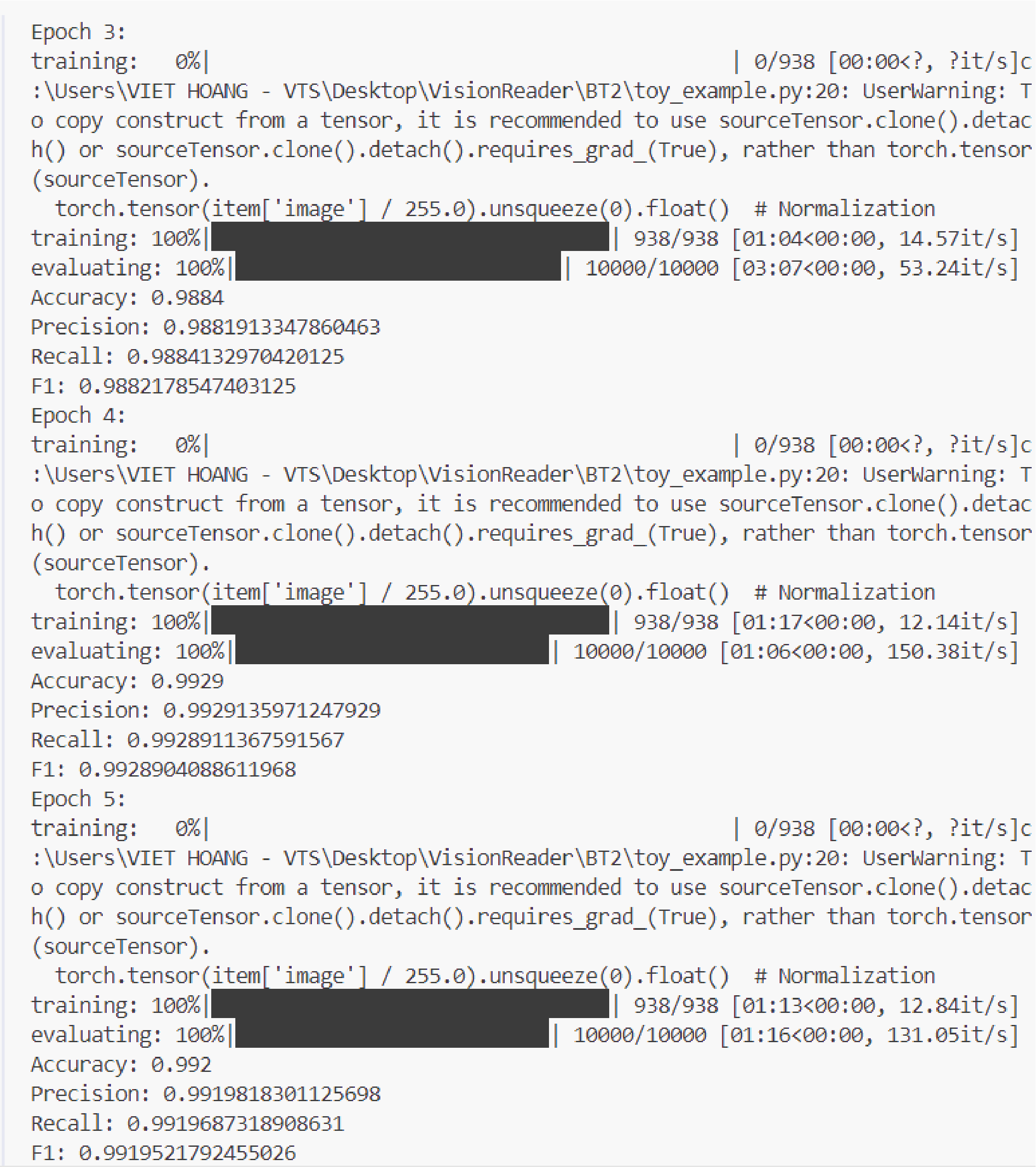


```
Epoch 10:
training:   0%|                                              | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|████████████████████████████████| 938/938 [00:25<00:00, 37.26it/s]
evaluating: 100%|█████████████████████████| 10000/10000 [00:59<00:00, 169.06it/s]
Accuracy: 0.9795
Precision: 0.9798237536030848
Recall: 0.9797064624222911
F1: 0.9796763844230634
```

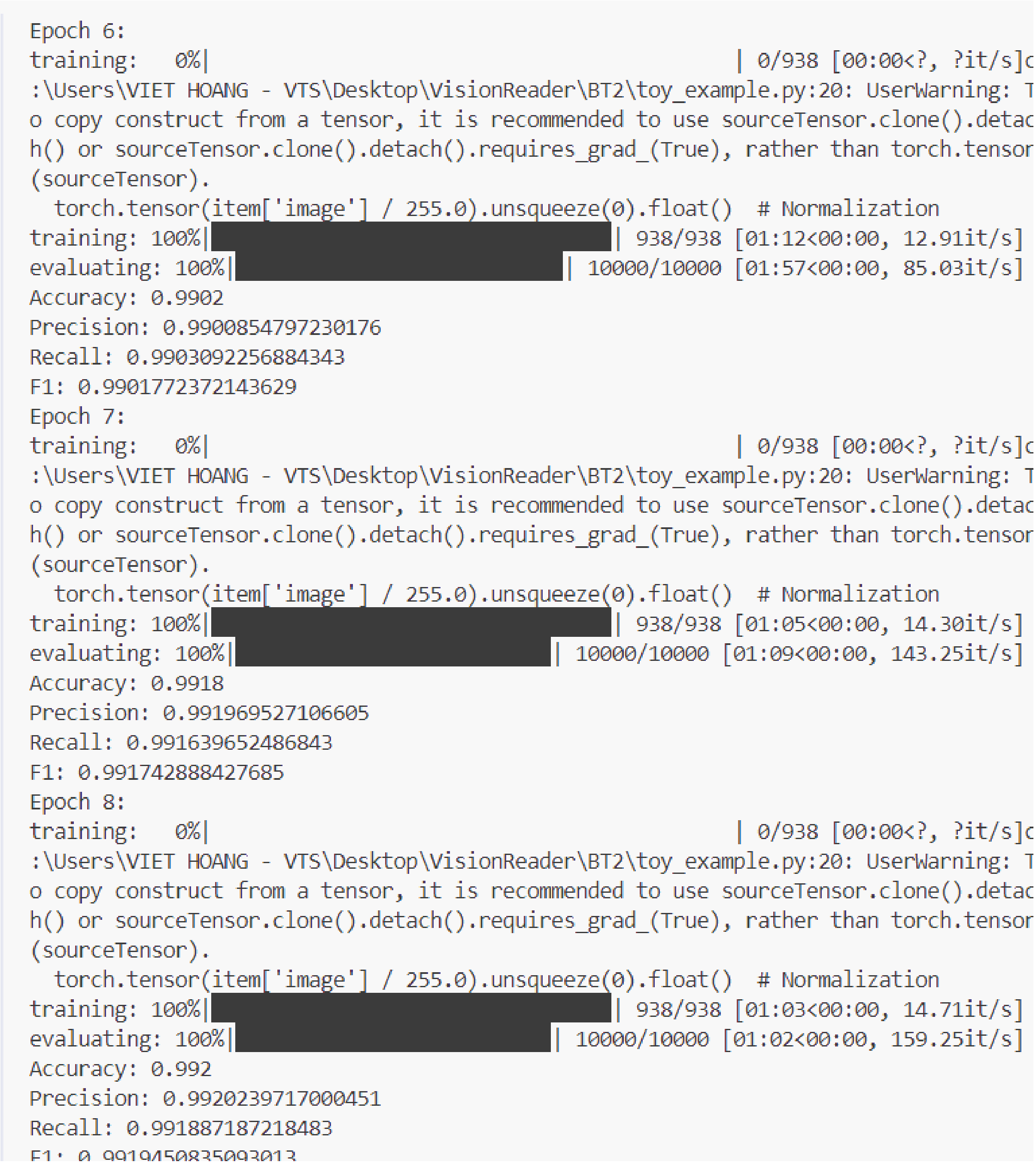
Câu 3:



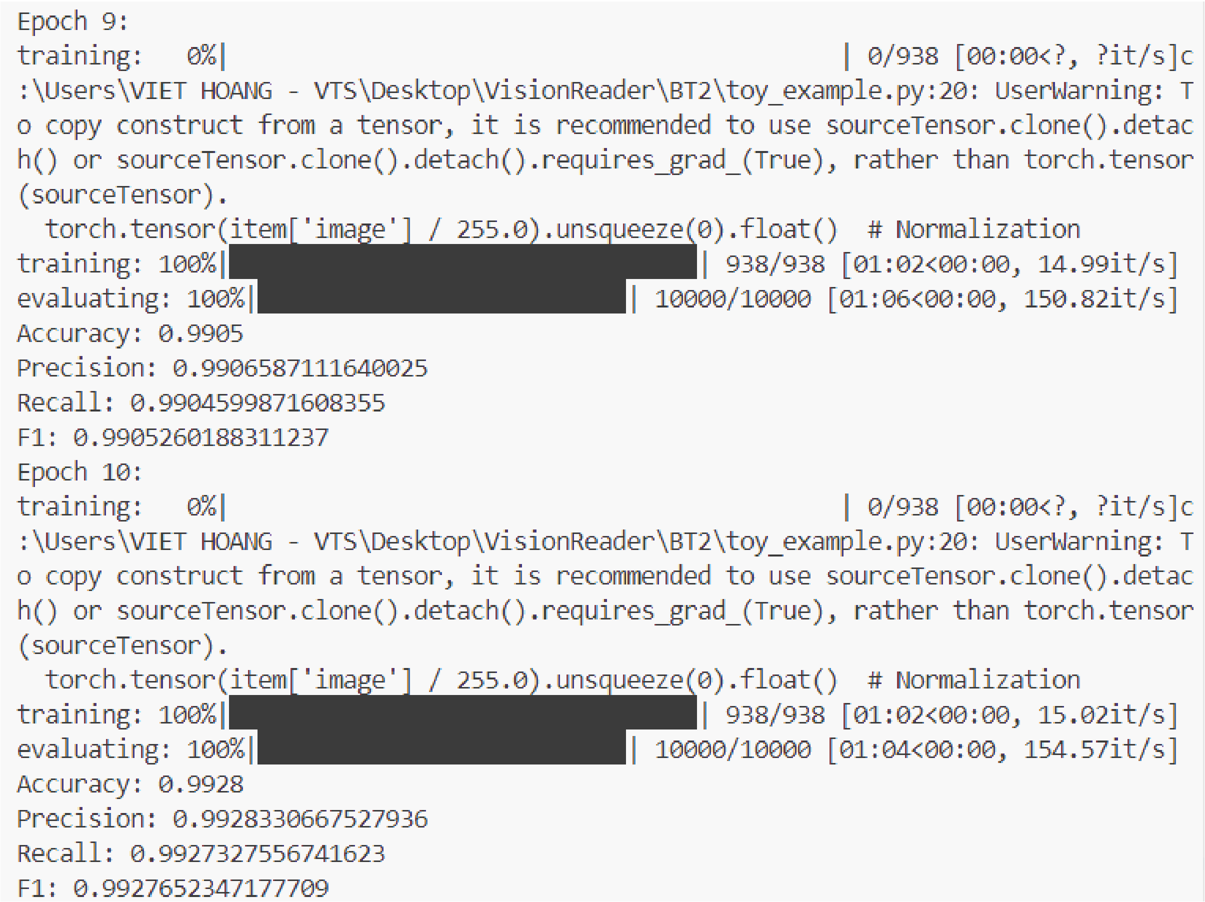
```
Epoch 1:
training:   0%|                                          | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [01:08<00:00, 13.70it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [01:00<00:00, 165.68it/s]
Accuracy: 0.9823
Precision: 0.9823605980778243
Recall: 0.9821499616479693
F1: 0.9821293945915182
Epoch 2:
training:   0%|                                          | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [01:01<00:00, 15.32it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [01:13<00:00, 136.64it/s]
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [01:01<00:00, 15.32it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [01:13<00:00, 136.64it/s]
Accuracy: 0.9905
Precision: 0.9905136253087556
Recall: 0.9905180591021091
F1: 0.9904927227635371
```



```
Epoch 3:
training:   0%|                                           | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [01:04<00:00, 14.57it/s]
evaluating: 100%|███████████████████████████| 10000/10000 [03:07<00:00, 53.24it/s]
Accuracy: 0.9884
Precision: 0.9881913347860463
Recall: 0.9884132970420125
F1: 0.9882178547403125
Epoch 4:
training:   0%|                                           | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [01:17<00:00, 12.14it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [01:06<00:00, 150.38it/s]
Accuracy: 0.9929
Precision: 0.9929135971247929
Recall: 0.9928911367591567
F1: 0.9928904088611968
Epoch 5:
training:   0%|                                           | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [01:13<00:00, 12.84it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [01:16<00:00, 131.05it/s]
Accuracy: 0.992
Precision: 0.9919818301125698
Recall: 0.9919687318908631
F1: 0.9919521792455026
```

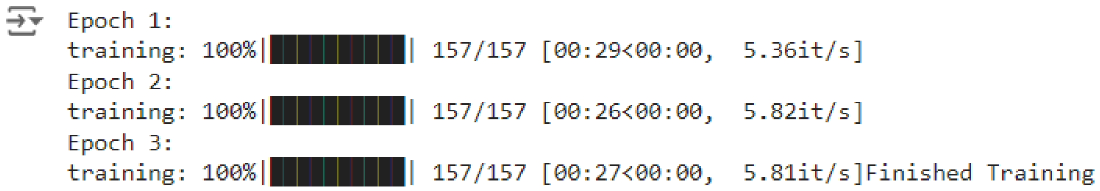


```
Epoch 6:
training:   0%|                                           | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [01:12<00:00, 12.91it/s]
evaluating: 100%|███████████████████████████| 10000/10000 [01:57<00:00, 85.03it/s]
Accuracy: 0.9902
Precision: 0.9900854797230176
Recall: 0.9903092256884343
F1: 0.9901772372143629
Epoch 7:
training:   0%|                                           | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [01:05<00:00, 14.30it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [01:09<00:00, 143.25it/s]
Accuracy: 0.9918
Precision: 0.991969527106605
Recall: 0.991639652486843
F1: 0.991742888427685
Epoch 8:
training:   0%|                                           | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|█████████████████████████████████| 938/938 [01:03<00:00, 14.71it/s]
evaluating: 100%|██████████████████████████| 10000/10000 [01:02<00:00, 159.25it/s]
Accuracy: 0.992
Precision: 0.9920239717000451
Recall: 0.991887187218483
F1: 0.9919450835093013
```



```
Epoch 9:
training:   0%|                                                   | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|██████████████████████████████████████| 938/938 [01:02<00:00, 14.99it/s]
evaluating: 100%|█████████████████████████████| 10000/10000 [01:06<00:00, 150.82it/s]
Accuracy: 0.9905
Precision: 0.9906587111640025
Recall: 0.9904599871608355
F1: 0.9905260188311237
Epoch 10:
training:   0%|                                                   | 0/938 [00:00<?, ?it/s]c
:\Users\VIET HOANG - VTS\Desktop\VisionReader\BT2\toy_example.py:20: UserWarning: T
o copy construct from a tensor, it is recommended to use sourceTensor.clone().detac
h() or sourceTensor.clone().detach().requires_grad_(True), rather than torch.tensor
(sourceTensor).
  torch.tensor(item['image'] / 255.0).unsqueeze(0).float()  # Normalization
training: 100%|██████████████████████████████████████| 938/938 [01:02<00:00, 15.02it/s]
evaluating: 100%|█████████████████████████████| 10000/10000 [01:04<00:00, 154.57it/s]
Accuracy: 0.9928
Precision: 0.9928330667527936
Recall: 0.9927327556741623
F1: 0.9927652347177709
```

Câu 4: Do e chạy trên colab lâu quá nên e chỉ run 3 epoch thui ạ :3



```
Epoch 1:
training: 100%|██████████| 157/157 [00:29<00:00,  5.36it/s]
Epoch 2:
training: 100%|██████████| 157/157 [00:26<00:00,  5.82it/s]
Epoch 3:
training: 100%|██████████| 157/157 [00:27<00:00,  5.81it/s]Finished Training
```