report

312552052鄭博元

overview

- 藉由bci使想像中的運動用信號處理來形成不同的EEG特徵
- 而CNN是目前用於分類的先進方式,因為它可以提取特徵,時間,空間等等提高分類性能,而內核則作為傳統時間和空間的過濾方式,經由一層又一層的feature提取最後能分成四個種類\

Detail implements--code

\\Dataloader中抓取data之feature及label藉由選模式載入train或test檔

\\Trainer檔中先利用reshape技巧使得吃進來的參數shape是正確的

而後定義loss跟accuracy的計算方式並且寫入torch_save使得最後訓練的權重會被鎖住寄存在另一個py檔,使用torch.adam增加模各方面性能

\\Test檔中先使用model.eval做評估,而測試中採用torch.no_grad使得test之權重不會更新在py檔,並且使用load_state_dict去做載入權重的動作

Detail implements--SCCNet

- 根據吃進來的數據有8個人每個人288筆test我們可以視為有2304個case,其中每個case根據倫文做法視為一個很大的平面而橫軸width為T(timesample)而因為為2D,所以我們定義z軸為1

  ==>Nu則依據倫ㄌ論文定義為22可視為其height
- 接著定義其kernel為size=Nc x Nt採用Nu作為filter數量對應
- 則下一層會產生以此convolution為基礎由Nu作為height的Nu x T之面積,論文中以20作為filter數量去對應kernel
- 最後我們經過非線性處理得到的權重值(也是以面構成)經過view一維化成一組組數據用stride一步一步移過去分成我們要的類別
- 有採取dropout跟square防止過擬和的問題

Experiment result

- 越高的epochs能確保我們收斂在正確的位置
- 學習率介於0.01能讓我們較不容易過度擬和使得test與train出的model非常不相容
- Batch數也會影響,一次抓多個程式能跑得較快,但也可能導致accuracy的不穩定,較容易泛化但梯度可能較不平穩不利於收斂

Experiment result—comparison(SD)

tester.py | SCCNet.py | trainer.py | Dataloader.py

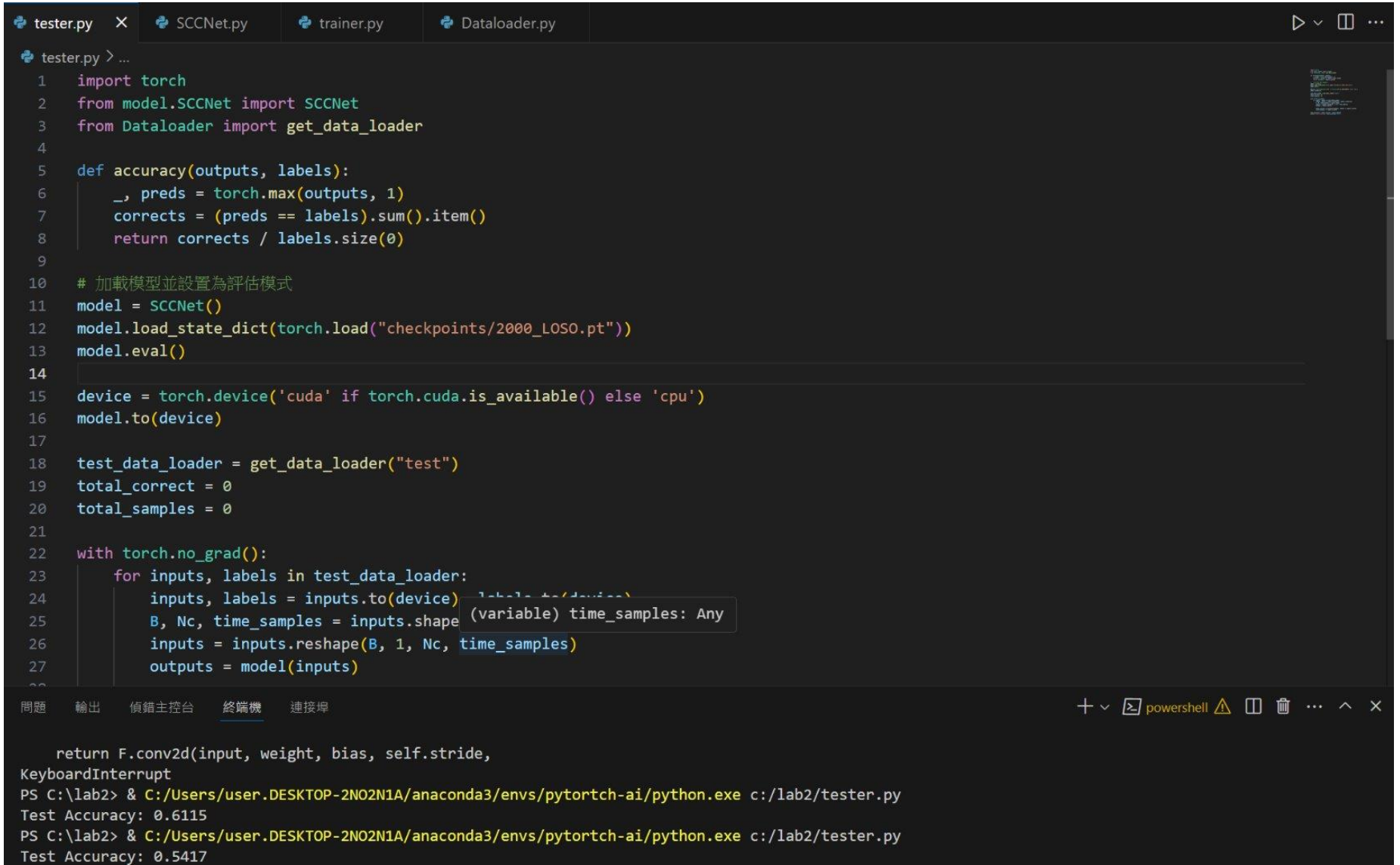
tester.py > ...

```python
import torch
from model.SCCNet import SCCNet
from Dataloader import get_data_loader

def accuracy(outputs, labels):
    _, preds = torch.max(outputs, 1)
    corrects = (preds == labels).sum().item()
    return corrects / labels.size(0)

# 加載模型並設置為評估模式
model = SCCNet()
model.load_state_dict(torch.load("checkpoints/2000_SD.pt"))
model.eval()

device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
model.to(device)

test_data_loader = get_data_loader("test")
total_correct = 0
total_samples = 0

with torch.no_grad():
    for inputs, labels in test_data_loader:
        inputs, labels = inputs.to(device), labels.to(device)
        B, Nc, time_samples = inputs.shape
        inputs = inputs.reshape(B, 1, Nc, time_samples)
        outputs = model(inputs)
```

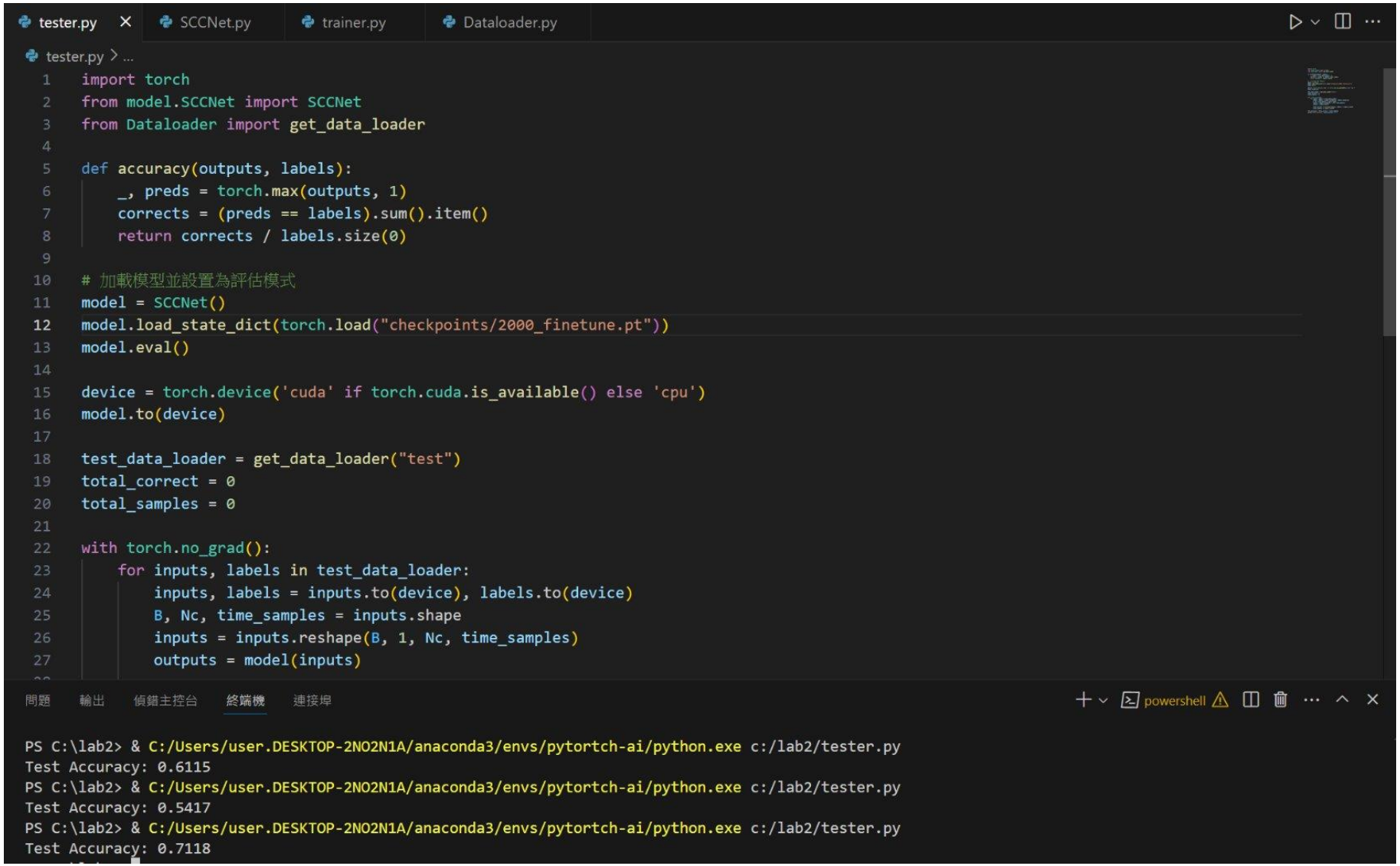
問題 輸出 偵錯主控台 終端機 連接埠 powershell

```
    return self._conv_forward(input, self.weight, self.bias)
  File "C:\Users\user.DESKTOP-2NO2N1A\anaconda3\envs\pytortch-ai\lib\site-packages\torch\nn\modules\conv.py", line 456, in _conv_forward
    return F.conv2d(input, weight, bias, self.stride,
KeyboardInterrupt
PS C:\lab2> & C:/Users/user.DESKTOP-2NO2N1A/anaconda3/envs/pytortch-ai/python.exe c:/lab2/tester.py
Test Accuracy: 0.6115
```

Experiment result—comparison(LOSO)



Experiment result—comparison(finetune)



Discussion

- 從噪聲較大的EEG信號中提取有意義的特徵並有效分類成不同的運動想象任務，需要先進的算法和相當的計算能力
- 且信號並非穩定性的也容易受到其他干擾影響準確率

- 實施像獨立成分分析（ICA）這樣的先進技術來去除EEG信號中的偽影
- 針對每個用戶的特定數據訓練模型，以考慮個體間的變異性。