HW2 辛普森人物辨識

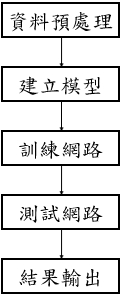
107368020 黃立

1. **作法說明：**

在Windows下建立TensorFlow與Keras模組，使用jupyter notebook來撰寫程式，程式是利用CNN卷積神經網路實現圖片判斷，以計算其辛普森人物辨識之精準度。

1. **程式方塊圖與寫法：**

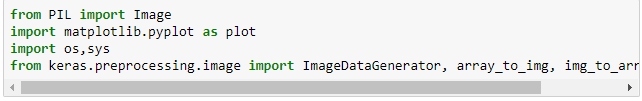
2.1 方塊圖：



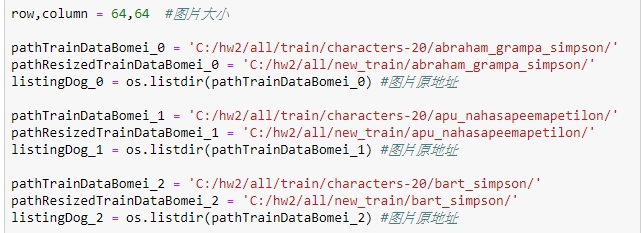
* 1. 程式寫法：

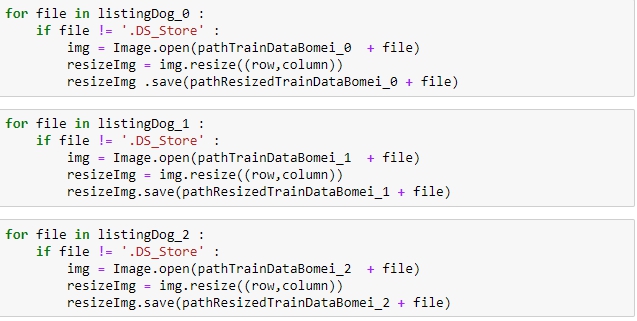
**在test\_1.ipynb檔案中**

Step1：import所需要的函式庫



Step2：選取轉換的圖片大小(64\*64)像素,並取得圖片原始資料夾中,再儲存新圖片至新資料夾中

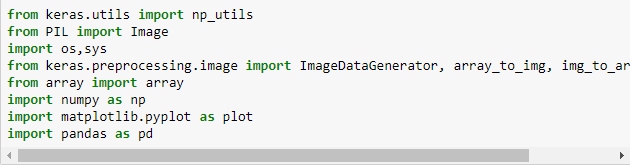




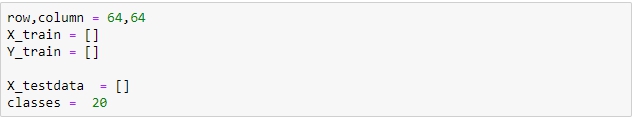


**在test3.ipynb的檔案中**

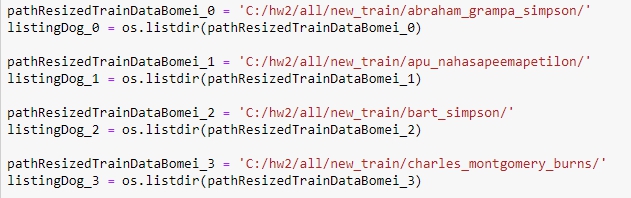
Step3：import所需要的函式庫

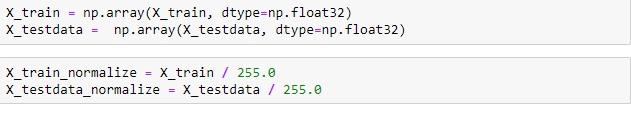


Step4：設定圖形大小,等待將圖片的RGB值,存入陣列中,並設定分成的類別數目

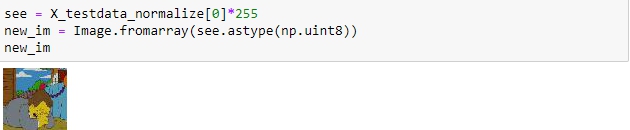


Step5：依序從資料夾讀取圖片,並給定標籤值

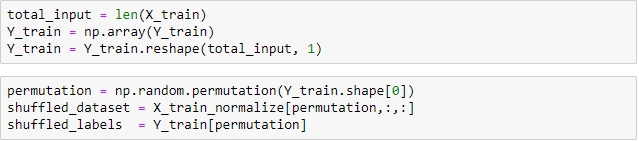


Step6：將圖片轉成array型態,並轉成浮點數

Step7：查看測試資料的圖片



Step8：隨機打亂訓練資料集的資料



Step9：將測試資料集的Label值轉成One-Hot encoding的型態

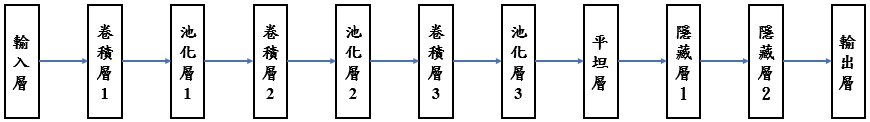


Step10：import訓練網路時所需要的函式庫

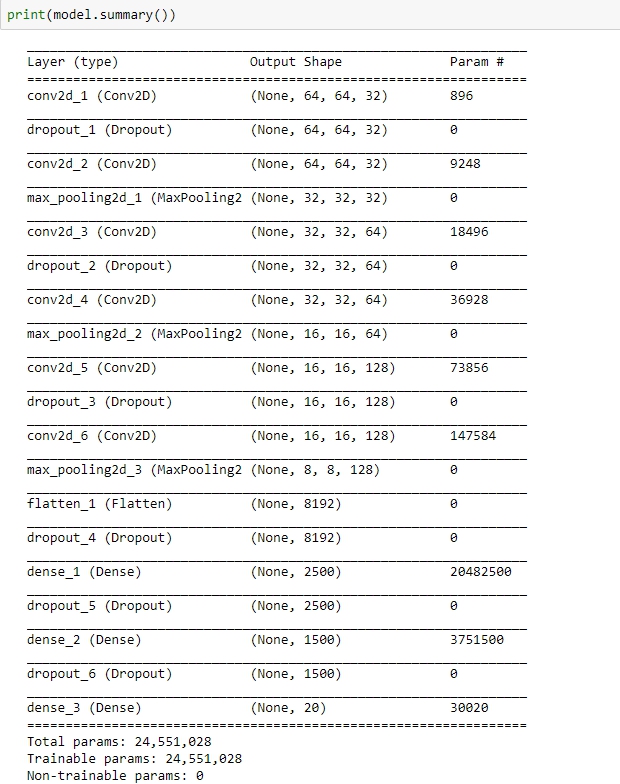


Step11：CNN網路模型





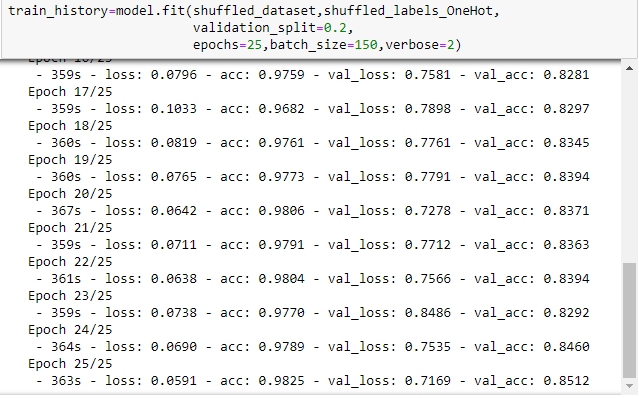
Step12：查看CNN網路模型架構



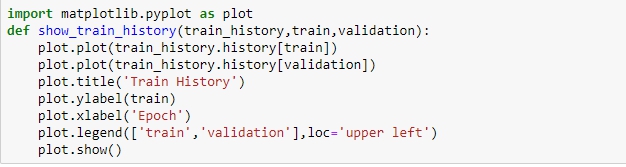
Step13：設定網路參數

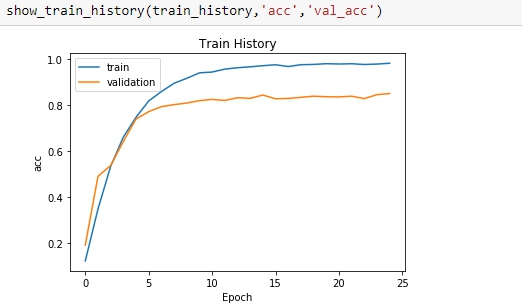


Step14：開始訓練網路



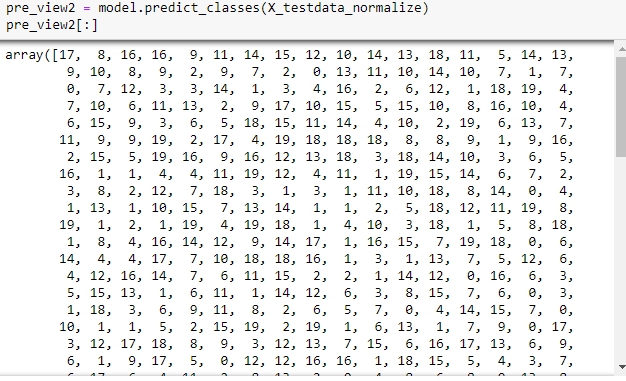
Step15：畫出Loss & Acc的圖查看訓練結果







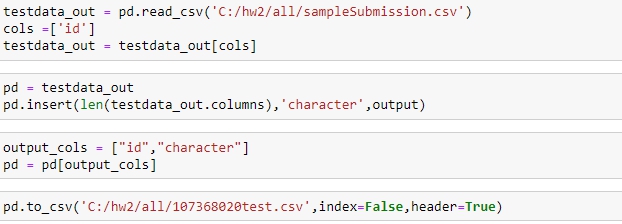
Step16：查看測試資料輸出結果值



Step17：設定字典型態以便轉換Label值

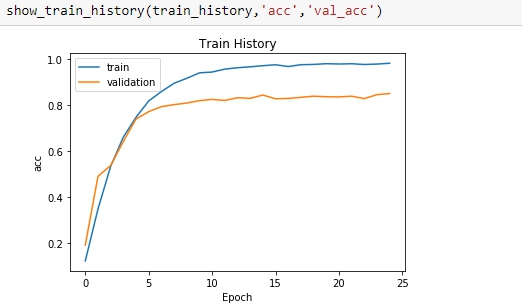


Step18：將輸出結果值,用DataFrame的形式整理好放入csv檔案中



1. **畫圖做結果分析:**

此實驗所預測之準確值與損失函數之圖形





1. **實驗中遇到之困難**
2. 第一次在測試時,訓練網路結果還不錯,但用來測試資料使用時,結果並不佳,因此查看了一下測試資料的輸入,才發現順序並沒有調整好,因此重新調整了輸入至網路的順序,才得到比較好的結果。
3. 在檔案匯入時,有一點小卡住,不知道怎麼正確的開啟圖片的位子,與轉換至相同的像素,還有要如何給定標籤值等的資料預處理不太熟悉,於是多查閱資料才知道該如何做。
4. **Kaggle排名**

