B0843026 資工二 謝奇容

程式流程圖：

MLP、CNN、RNN三種模型大致上都是以這個流程去執行，但細部會根據模型的需求，有不一樣的過程。

引入套件、資料集

資料前處理

設置模型

訓練模型

測試模型

程式解說及執行結果：

1. MLP
   1. 引入需要使用的套件及資料集，並設定資料集一張含有 文字 的圖片

      自動產生的描述
   2. 接著，做資料的前處理一張含有 文字 的圖片

      自動產生的描述

圖片標準化，將圖片特徵設於0~1之間

將圖片轉成矩陣

* 1. 再來設置模型一張含有 文字 的圖片

     自動產生的描述程式執行結果
     自動產生的描述

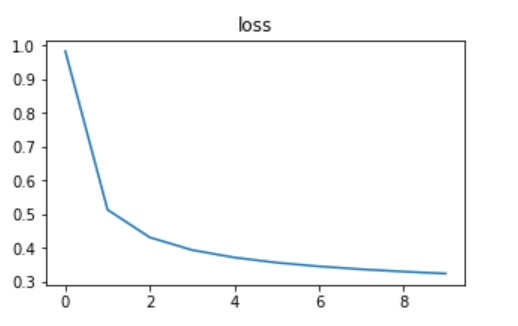
設置模型的全連接層

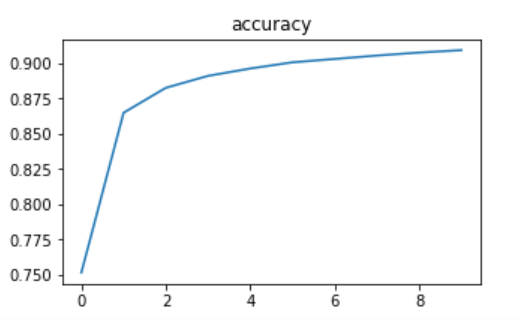
* 1. 訓練模型一張含有 文字 的圖片

     自動產生的描述程式執行結果
     自動產生的描述

繪製準確率圖形

繪製損失函數圖形

* 1. 程式執行結果

1. CNN
   1. 引入套件及資料集一張含有 文字 的圖片

      自動產生的描述
   2. 接著做資料的前處理一張含有 文字 的圖片

      自動產生的描述

圖片標準化，將圖片特徵設於0~1之間

將圖片轉換成符合圖片的4維矩陣

*Onehot encoding是一個二維矩陣，用0/1來表示沒有序列大小的離散資料*

* 1. 再來設置模型一張含有 文字 的圖片

     自動產生的描述

避免過度訓練

設置兩個卷積層及兩個池化層

*relu：小於0的數值隱藏；卷積後圖片會變小，為了不要變小，使用"same"函數，在卷積前，匯在原圖片外面補0，這樣卷積後圖片就不會變小了*一張含有 桌 的圖片

自動產生的描述

* 1. 訓練及測試模型一張含有 文字 的圖片

     自動產生的描述一張含有 桌 的圖片

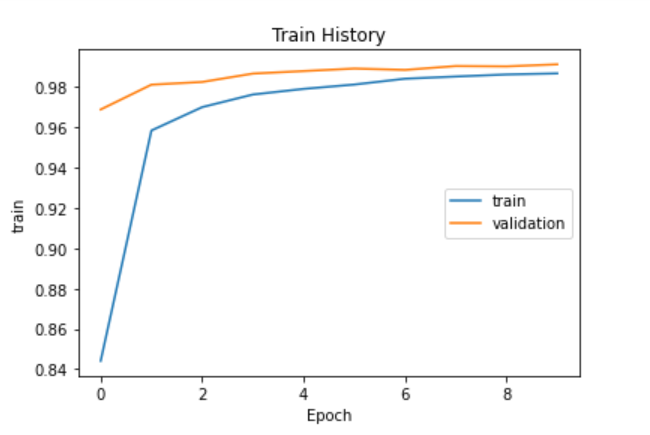
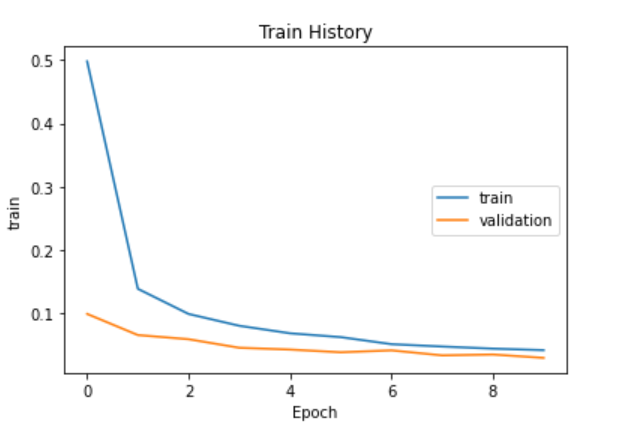
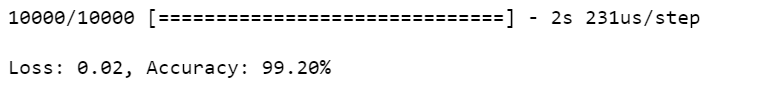
     自動產生的描述

顯示模型預判情況

繪製損失函數圖形

繪製準確率圖形

訓練模型

* 1. 程式執行結果一張含有 桌 的圖片

     自動產生的描述

1. RNN
   1. 引入套件及資料集一張含有 文字 的圖片

      自動產生的描述
   2. 做資料前處理一張含有 文字 的圖片

      自動產生的描述

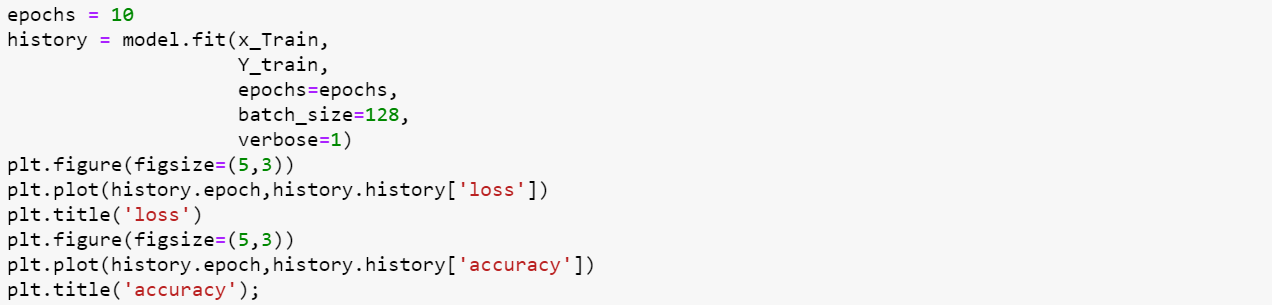
圖片標準化，將圖片特徵設於0~1之間

* 1. 設置模型一張含有 文字 的圖片

     自動產生的描述一張含有 桌 的圖片

     自動產生的描述

設置輸出層

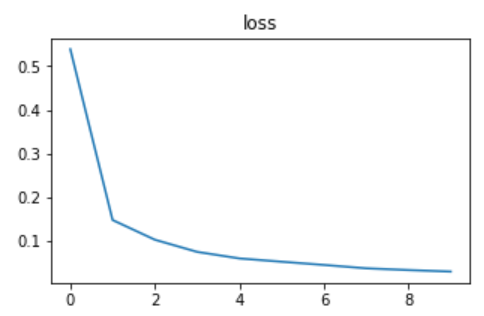
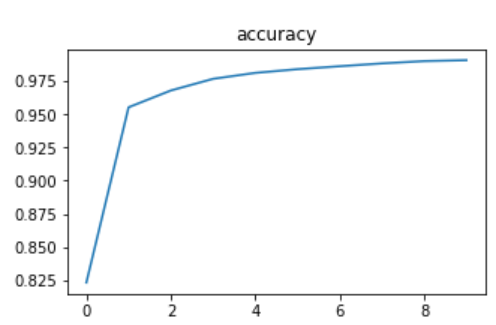
* 1. 訓練模型及測試模型一張含有 桌 的圖片

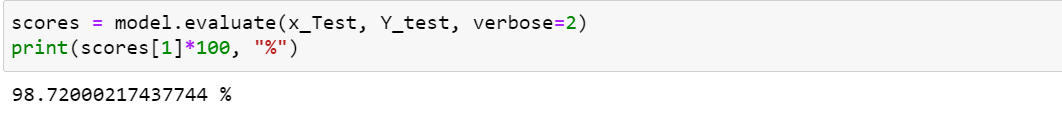
     自動產生的描述

繪製準確率圖形

繪製損失函數圖形

訓練模型

* 1. 程式執行結果



程式執行遇到的問題：

　　第一次執行時，顯示的損失函數不高，準確率不低，最後的準確率也都有超過90，但不知道為什麼，最後三個方法畫出來的圖，圖形長得很奇怪，圖形上的數值很低，導致線的斜率很奇怪。不過，我也沒有改動什麼程式碼，就只是重新執行，執行2次後，三個方法的準確率都有提升，圖形也呈現出跟數值相符的樣子了。這個情況跟比賽時遇到的問題有點像，每次執行深度學習的模型建構、訓練的程式後，都會產省不太一樣的結果，大部分的時候只會鄉差一點點，但偶而會跑出相差很大的結果，這時候重新執行程式，可能就能解決問題了。