**程式語言-期末專題報告（第12組）**

1. **成員:**

組長：資工2B 108502571 楊佳峻

組員：資工2B 108502570 吳榕憲

組員：資工2B 108502572 吳秉鴻

1. **題目：GAN手寫數字生成**
2. **實作方法：**

使用Python3.8、TensorFlow套件

網站：<https://bit.ly/3lAsvUe>

參考論文方法說明:

GAN可以看成是一個框架，它包含一個generator和discriminator。Generator負責透過模型輸入，生成能夠媲美真實數據的結果；discriminator則負責防偽，也就是能辨識generator的輸出與真實數據。在兩種模型的對抗下

，generator的輸出可以越來越逼近真實數據，而discriminator則越來越精明，能辨識細微的真偽。

1. **GAN生成模型與辨識模型架構**

**一張含有 文字 的圖片

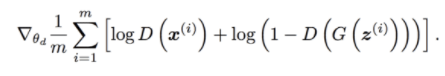
自動產生的描述Discriminator:** conv2d作為卷積層取得圖像的特徵值，batch normalization防止梯度消失並加快訓練速度，使用leaky relu作為激勵函數 max pooling將節點數縮小，重複兩次將圖片從28\*28變成7\*7 使用flatten將資料變為一維，最後縮小變成1\*11的扁平矩陣以表達圖片答案為各數字的可能性，所使用的批次為64。

**一張含有 文字 的圖片

自動產生的描述一張含有 文字 的圖片

自動產生的描述Generator:** input為1\*10的one hot矩陣及1\*100的雜訊，一樣使用batch nomalization防止梯度消失並使用leaky relu作為激勵函數，透過reshape將矩陣變為7\*7的圖形，透過up sampling擴大圖片，使用conv2d transpose(反卷積)取樣，重複兩次可得28\*28的圖片。

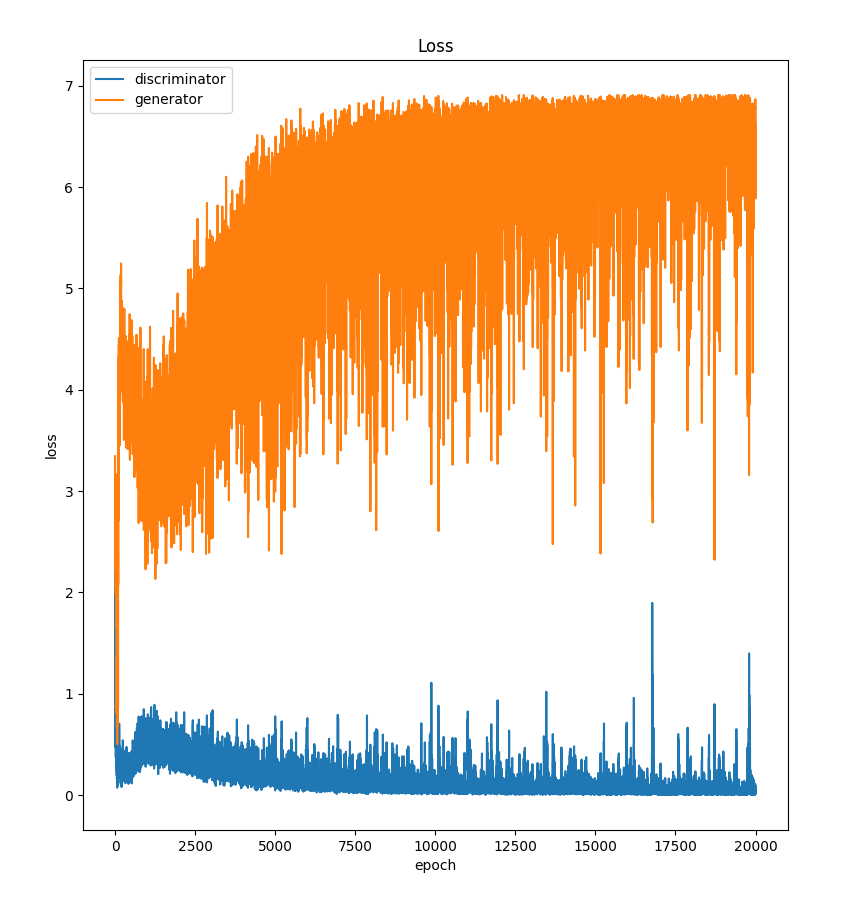
1. **Loss function**

**Discriminator:**

**一張含有 文字, 手錶, 量表 的圖片

自動產生的描述**

**Generator:**

* X: 批次為i的mnist資料
* G(z) : generator產生的批次為i的資料
* D() : discriminator給出正確標籤的機率
* discriminator : 當輸入資料為mnist時使輸出結果為正確標籤，而輸入資料為 generator產生的資料時，使輸出結果為錯誤標籤
* generator : 使discriminator判斷其產生的資料時輸出結果為正確標籤
* loss function曲線圖

1. **生成手寫數字(訓練10000次)**

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

1. **個人心得**

**楊佳竣:** Gan生成對抗網路真的很有趣，理論上，透過兩個神經網路可以互相強化對方，但在實作上，卻要花不少心力在連接他們，不過，成果是好的，的確值得花時間常識，而老師教的Matlab畫圖法也與Python語法十分雷同，很好上手，謝謝老師這學期的教導，讓我從零開始學會一種程式語言。

**吳榕憲:** 這次的專題讓我學習如何自學，自己找網路資源學習如何實作生成對抗網路，根據教學打造網路並除錯。也讓我找到google colab加速結果生成。此後若要再構造別的深度學習模型必能活用此次的結果。

**吳秉鴻:** 這次的GAN手寫數字生成讓我學到了生成對抗網路的結構，在訓練模型的時候也學到了如何使用google colab的GPU加速系統，在demo的時候，了解到loss function在人工智慧模型中佔有非常重要的地位，雖然我們生成的數字還有待加強，但是我覺得這一學期學到的非常多，經過這次的經驗，以後可以避免一些錯誤的方法，使結果更好看一些。