MLDS HW4 Report

電機四 B03901030 蕭晨豪

1. Model description

我的模型使用的是一般的 DC GAN+ skip thought vectors，首先我將文字轉成 4800 維的 skip thought vector，然後在 generator 和 discriminator 均訓練一個 embedding，將這個 4800 維 map 到 128 維。

將 generator 的 input 設為 100 維的 gaussian noise 和 128 維的 concat，然後經過如下模型輸出一張 64x64x3 的圖片:

Deconv(100+128,512,kernel = 4x4, stride = 1x1)
BatchNormalization
Deconv(512,256,kernel = 4x4, stride = 2x2, padding = 1x1)
BatchNormalization
Deconv(256,128,kernel = 4x4, stride = 2x2, padding = 1x1)
BatchNormalization
Deconv(128,64,kernel = 4x4, stride = 2x2, padding = 1x1)
BatchNormalization
Deconv(64,3,kernel = 4x4, stride = 2x2, padding = 1x1)

中間的 deconvolution 層都 Relu，並且最後一層的 output 經過 tanh 函數

Discriminator 的模型架構如下：

Conv(3,64,kernel = 4x4, stride = 2x2, padding = 1x1)
BatchNormalization
Conv(64,128,kernel = 4x4, stride = 2x2, padding = 1x1)
BatchNormalization
Conv(128,256,kernel = 4x4, stride = 2x2, padding = 1x1)
BatchNormalization
Conv(256,512,kernel = 4x4, stride = 2x2, padding = 1x1)
BatchNormalization

在這裡將 128 維的文字 vector 複製四份，和 convolution 出來的結果 concat，然後再繼續添加 convolution layer

Conv(512+128,1,kernel = 4x4, stride = 1x1)

最後通過一個 sigmoid，其中上面的 convolution layer 均在 batch normalization 後通過一個斜率為 0.2 的 leaky relu

在訓練時我將 Discriminator 的 loss 分為四個部分計算:

首先 real img, right text 與全 1 的 vector 計算 BCE(Binary Cross Entropy)，因為我們希望 discriminator 將他視為正確的。

然後 wrong img, right text 和 real img, wrong text 與全 0 的 vector 計算 BCE，這裡的 wrong 指的是從原本的 dataset sample 出與 batch data 不同的資料。最後 fake img(generator 產生出來的),right text 與全 0 的 vector 計算 BCE。將第一個與後三個的平均加起來便是 discriminator 的 loss

Generator 的 loss 則相當簡單，即為其生成出的圖片通過當時的 discriminator 所得到的分數與全 1 的 vector 計算 BCE，因為我們希望它能騙過 discriminator。

2. Improve performance

用上述方法訓練的 generator 不是非常理想，我作了以下幾個 improvement:

(1) Data augmentation (copy) :這裡我並沒有找額外 data，甚至也並沒有做翻轉或旋轉，主要是我發現我訓練出來的 generator 對於只有髮色或只有眼睛顏色的 description 效果非常差，description 必須要包含髮色「和」眼睛顏色兩個 tag 才能生成出較像人臉的圖像，因此我在 training 時讓有兩個 tag 的圖片有 0.3 的機率被複製然後將它的 tag 分開，舉例來說若原本有一筆資料形如{圖片 A，'blue hair green eyes'}，則它有 0.3 的機率變成三筆資料{圖片 A，'blue hair green eyes'},{圖片 A，'green eyes'},{圖片 A，'blue hair'}，透過這種方式來增加只有一個 tag 的資料，最後在只有一個 tag 的敘述上生成的圖片也有所進步。

(2) One sided label smoothing :將正確的 data 的分數設為 0.9，因此 discriminator 在訓練時 real img, right text 的分數是與全部都是 0.9 的 vector 計算 BCE。

(3) Feature matching : 在計算 generator loss 時增加一個從 convolution layer 抽出來的 feature 的 L2 loss

3. Experiment setting and observation

我主要觀察到的現象是：一開始使用 4800 維的 skip thought vector，一般的 GAN 在 epoch 數較多時容易發生 mode collapse 問題，在 300 個 epoch 時同一個敘述產生的都是同一個圖片了，而我對此進行了將 4800 維降到 128,256,512 的實驗，發現降到越低維度時 mode collapse 問題減少的越顯著，因此最後選用 128 維作為我最終的 model。