

Machine Learning (2017, Fall)

Assignment 3 – Image Sentiment Classification

學號：b03901086 系級：電機四 姓名：楊正彥

1 (1%) Build Convolution Neural Network

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 32, 48, 48)	320
dropout_1 (Dropout)	(None, 32, 48, 48)	0
conv2d_2 (Conv2D)	(None, 64, 48, 48)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 64, 24, 24)	0
dropout_2 (Dropout)	(None, 64, 24, 24)	0
conv2d_3 (Conv2D)	(None, 64, 24, 24)	36928
dropout_3 (Dropout)	(None, 64, 24, 24)	0
conv2d_4 (Conv2D)	(None, 128, 24, 24)	73856
max_pooling2d_2 (MaxPooling2D)	(None, 128, 12, 12)	0
dropout_4 (Dropout)	(None, 128, 12, 12)	0
conv2d_5 (Conv2D)	(None, 128, 12, 12)	147584
dropout_5 (Dropout)	(None, 128, 12, 12)	0
conv2d_6 (Conv2D)	(None, 256, 12, 12)	295168
max_pooling2d_3 (MaxPooling2D)	(None, 256, 6, 6)	0
conv2d_7 (Conv2D)	(None, 256, 6, 6)	590080
dropout_6 (Dropout)	(None, 256, 6, 6)	0
flatten_1 (Flatten)	(None, 9216)	0
dropout_7 (Dropout)	(None, 9216)	0
dense_1 (Dense)	(None, 1024)	9438208
dropout_8 (Dropout)	(None, 1024)	0
dense_2 (Dense)	(None, 7)	7175
Total params: 10,607,815		
Trainable params: 10,607,815		
Non-trainable params: 0		

此次作業所實作的 CNN 模型架構圖如左，包含了四個主要的 conv blocks，每一層 conv blocks 都由兩層的 Conv2D、兩層的 Dropout 以及一層的 MaxPooling2D 所組成，再經過 flatten 以及兩層的 dense 以及 dropout，最後由 softmax 得出我們的 prediction 結果。整個 CNN 模型的 parameters 總數為 10,607,815 個。

至於 preprocessing 的部分，我先是把一些壞掉的 image（如全黑、只有符號等）去除，並對所有的 training data 做標準化。接下來用了 keras 內建的 ImageDataGenerator 套件對 training data 進行平移、旋轉、縮放、水平翻轉等處理已產生更多的 training data，以達到減少 over-fitting 的效果，val-acc 也從 0.66 上升到了 0.70 左右。

而其他訓練相關參數為：batch size=512、training epoch=400、loss 和 optimizer 分別採用 cross entropy 以及 lr=0.00027 的 Adam。

除此之外也有透過實作 ensemble learning 的方式訓練模型，一共使用了 10 個左右不同架構的 CNN 模型做合併，但是最後 val-acc 約為 0.71 左右並沒有太顯著的提升，故最後作業上傳的版本仍為使用單一模型的 CNN 架構。

從 Fig 1 的 CNN Model Accuracy 訓練過程可以看出來，其實模型大概在 50 個 epochs 左右後就已經開始收斂，最好的 val-acc 出現在約 300 個 epochs 附近。同時也可以觀察到 training accuracy 的震盪幅度遠比 validation accuracy 要來的小很多。

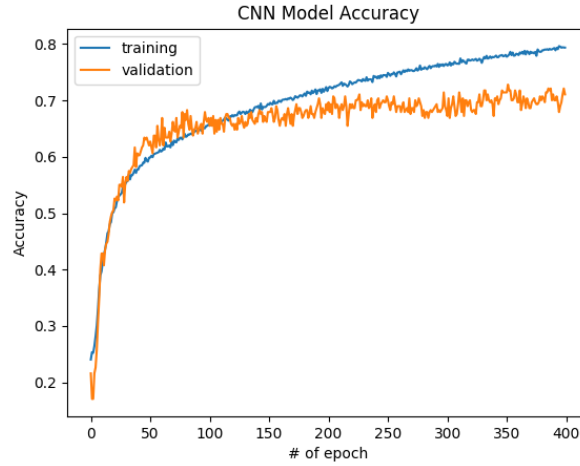


Fig 1: Training/Validation Accuracy for each epoch of CNN

2 (1%) Build Deep Neural Network

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 1024)	2360320
dense_2 (Dense)	(None, 512)	524800
dense_3 (Dense)	(None, 512)	262656
dense_4 (Dense)	(None, 256)	131328
dense_5 (Dense)	(None, 256)	65792
dense_6 (Dense)	(None, 128)	32896
dense_7 (Dense)	(None, 64)	8256
dropout_1 (Dropout)	(None, 64)	0
dense_8 (Dense)	(None, 7)	455
Total params: 3,386,503		
Trainable params: 3,386,503		
Non-trainable params: 0		

此次作業所實作的 DNN 模型架構圖如左，主要是參考網路上一般 DNN 架構並將 parameters 總數調整到與上一題 CNN 模型有著同樣數量級，一共是 3,386,503 個。

從 Fig 2 可以發現，在相同的資料處理以及相關訓練參數的選擇下 DNN 模型的準確度遠不及 CNN 模型所得到的結果，而 training-acc 也很容易在少量的 epochs 之後達到相當嚴重的 over-fitting 問題，可能是因為 DNN 模型不像 CNN 模型能夠透過 Conv2D

layer 來學習到這些圖片的局部特徵，所以 performance 才會無法與 CNN 相提並論。

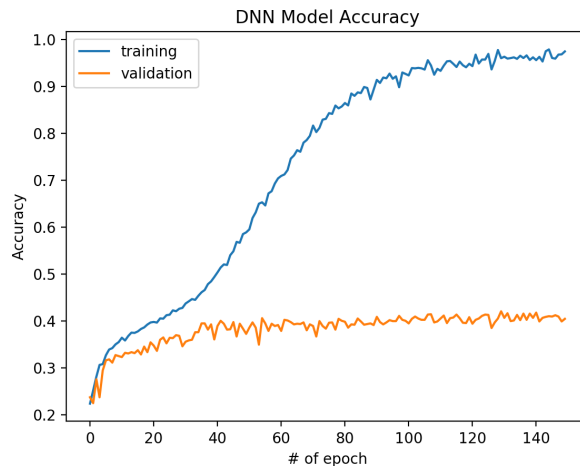


Fig 2: Training/Validation Accuracy for each epoch of DNN

3 (1%) Analyze the Model by Confusion Matrix

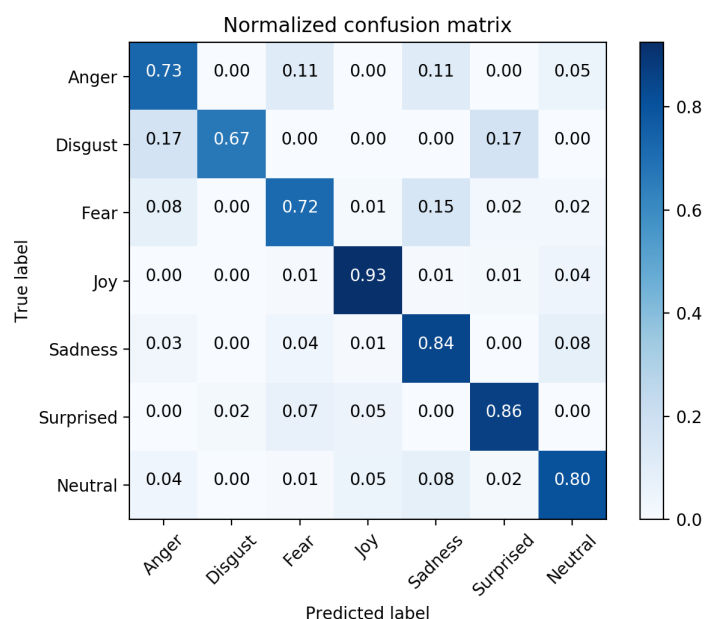


Fig 3: Normalized Confusion Matrix on validation data

從 Confusion Matrix 可以發現 Disgust、Fear、Anger 依序為這次實作之 CNN 模型最難成功預測前三個 sentiment class，而預測正確率最高的則為 Joy，有可能是因為 Joy 與其他 sentiments 相差最多。而其他容易混淆的組合包括把 Disgust 判斷成 Surprised、把 Disgust 判斷 Anger 以及把 Fear 判斷成 Sadness。

4 (1%) Analyze the Model by Plotting the Saliency Map

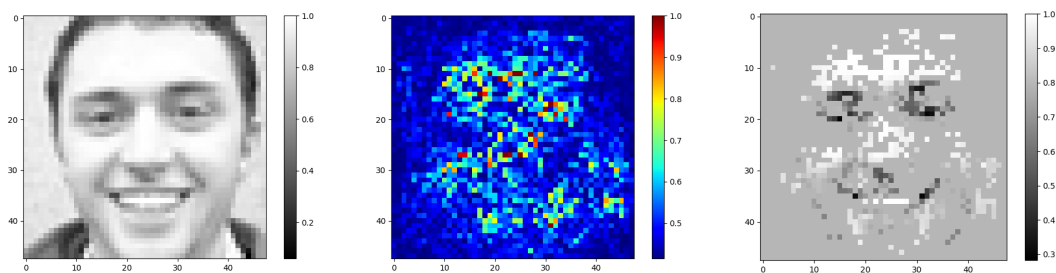


Fig 4: Images of Original/Heatmap/Masked Testing Data

從 Fig 2 可以看到當我們把原圖的 Saliency Map 還原並且經過 normalization 之後，經過與 masked image 的比較可以發現 heatmap 中在人臉中眼睛和嘴巴周圍部分有比較高的數值，可以知道我們的 CNN 模型在做 sentiment classification 的時候將高度依賴人臉中眼睛和嘴巴周圍的特徵來做出預測及判斷。

5 (1%) Analyze the Model by Visualizing Filters

Fig 5 和 Fig 6 為我 visualize 模型中第二層與第六層 conv2D 中 filter 做 180 個 epochs 的 gradient ascent 所得到的結果，可以觀察發現其紋理有越來越細緻越複雜的狀況，但是還是難以推斷出這些和人臉表情之間的關係，而 Fig 7 為將 id=5566 的 image 丟進 CNN 模型 conv2d_2 層所得到各 layer 的 outputs。

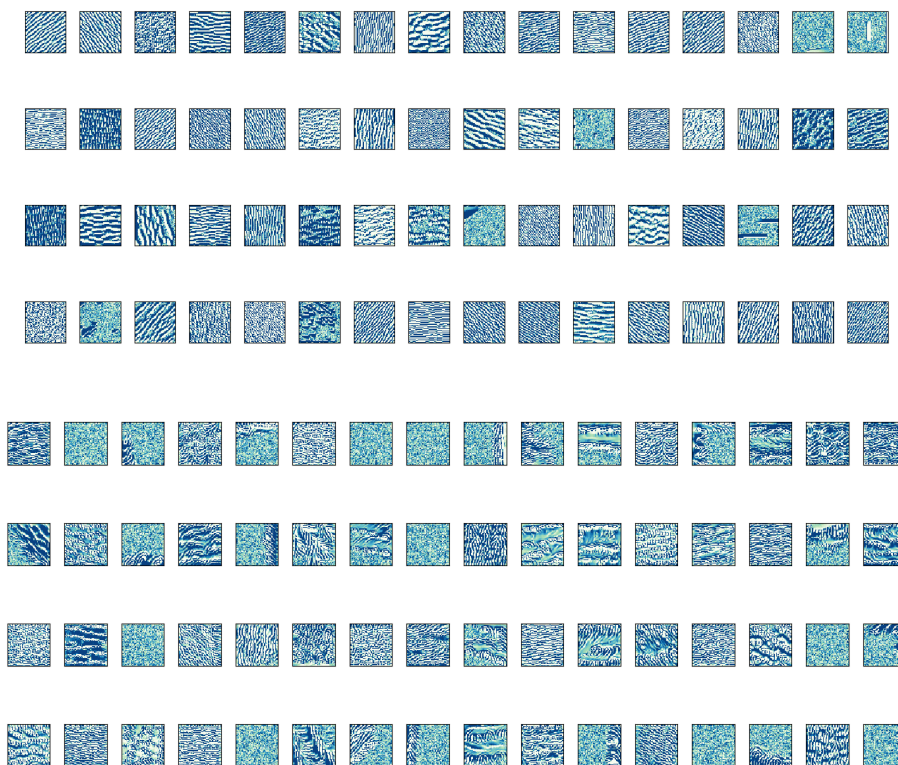


Fig 5: Filters of Layer – conv2d_2 and conv2d_6 with Ascent Epochs = 180

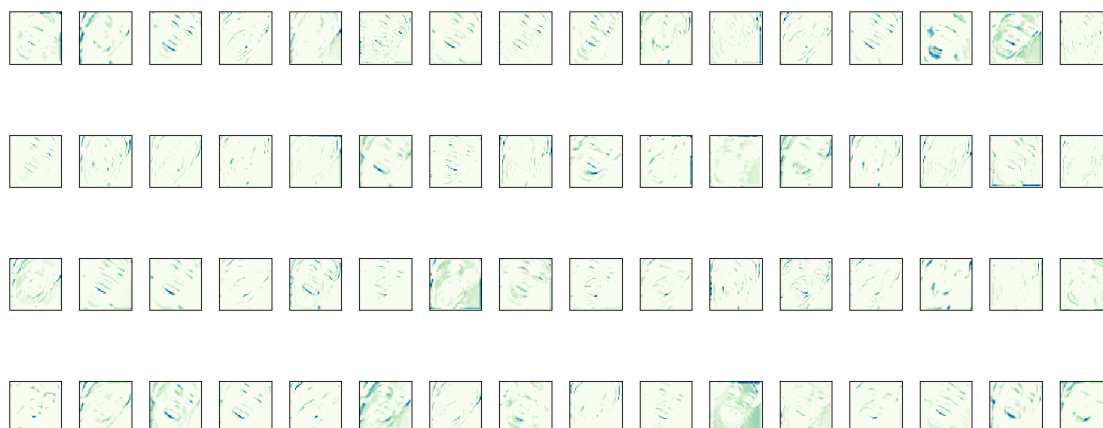


Fig 6: Output of Layer – conv2d_2