

# Homework 3 Report - Image Sentiment Classification

學號：r06921081 系級：電機碩一 姓名：張邵瑀

1. (1%) 請說明你實作的 CNN model，其模型架構、訓練參數和準確率為何？

vgg-16改		vgg-16 mini		改進小老師模型	
Conv2D 64x(3x3) Acti: relu Drop:0.10 Conv2D 64x(3x3) Acti: relu Drop:0.10 MaxPooling2D 2x2 Drop:0.10  Conv2D 128x(3x3) BatchNormalization() Acti: relu Drop:0.25 Conv2D 128x(3x3) Acti: relu Drop:0.25 MaxPooling2D 2x2 Drop:0.25  Conv2D 256x(3x3) Acti: relu Drop:0.35 Conv2D 256x(3x3) Acti: relu Drop:0.35 Conv2D 256x(3x3) BatchNormalization() Acti: relu Drop:0.35 MaxPooling2D 2x2 Drop:0.35  Conv2D 512x(3x3) Acti: relu Drop:0.4 Conv2D 512x(3x3) Acti: relu Drop:0.4 Conv2D 512x(3x3) Acti: relu Drop:0.4 MaxPooling2D 2x2 Drop:0.4  Conv2D 512 x(3x3) Acti: relu Drop:0.45 Conv2D 512 x(3x3) Acti: relu Drop:0.45 Conv2D 512 x(3x3) Acti: softplus Drop:0.45 MaxPooling2D 2x2 Drop:0.45  Flatten() Dense 4096 Acti: relu Drop:0.5 Dense 4096 Acti: softplus Drop:0.5 Dense 7 Acti: softmax		Conv2D 64x(3x3) Acti: relu Conv2D 64x(3x3) Acti: relu MaxPooling2D 2x2 Drop:0.10  BatchNormalization() Conv2D 128x(3x3) Acti: relu MaxPooling2D 2x2 Drop:0.25  Conv2D 256x(3x3) BatchNormalization() Acti: relu MaxPooling2D 2x2 Drop:0.35  Conv2D 512x(3x3) Acti: relu MaxPooling2D 2x2 Drop:0.4  Conv2D 512x(3x3) Acti: relu Conv2D 512x(3x3) Acti: softplus MaxPooling2D 2x2 Drop:0.45  Flatten() Dense 4096 Acti: relu Dense 4096 Acti: softplus Drop:0.5 Dense 7 Acti: softmax		Conv2D 64x(5x5) LeakyReLU:0.05 BatchNormalization MaxPooling2D (2x2) Drop:0.5  Conv2D 128x(3, 3) LeakyReLU:0.05 BatchNormalization MaxPooling2D (2x2) Drop:0.5  Conv2D 512x(3x3) LeakyReLU:0.05 BatchNormalization MaxPooling2D (2x2) Drop:0.5  Conv2D 512x(3x3) LeakyReLU:0.05 BatchNormalization MaxPooling2D (2x2) Drop:0.5  Flatten() Dense 512 Acti:relu Drop:0.5 Dense 512 Acti:softplus BatchNormalization Drop:0.5 Dense 7 Acti:softmax	
optimizer: adam lr = 1e-4 batchsize=128	ReduceLROnPlateau-factor: 0.1 patience:25	optimizer: adam lr = 1e-4 batchsize=128	ReduceLROnPlateau-factor: 0.1 patience:25	optimizer: adam lr = 1e-4 batchsize=128	ReduceLROnPlateau-factor: 0.1 patience:25
public:0.68904	private:0.68403	public:0.68292	private:0.68069	public:0.71022	private:0.70799

討論：一開始用VGG16的模型下去嘗試，接著嘗試增加Dropout層，加法為設定一開始的conv層較重要需分辨最粗略的feature Drop率低，效果非常顯著，後來試著改變activation函數，發現在後層使用softplus效果也有所提升，VGG做到最好是0.689，但發現訓練時間太久所以減少層數發現準確率也沒有降多少，在early deadline 結束後助教跟小老師說LeakyReLU跟PReLU不錯，但我發現PReLU反而更糟，所以後來只有留下LeakyReLU跟我自己前面測出的Dropout跟softplus並把其他BatchNormalization加上，就得到single model 0.71022的成績，之後就沒在上去過了，如果有用ensemble到0.73應該也不是問題，不過助教禁止所以就放棄了。

2. (1%) 請嘗試 data normalization, data augmentation, 說明實行方法並且說明對準確率有什麼樣的影響？

**data normalization:**

先減掉mean(all data)，然後做 normalization，有兩種方式：

1. 一種直接在讀完檔之後將該變數直接除以255
2. 用keras ImageDataGenerator 中設定rescale=1./255 (我採用這種)

	實作前	實作後
acc	0.65645	0.68904

**data augmentation:**

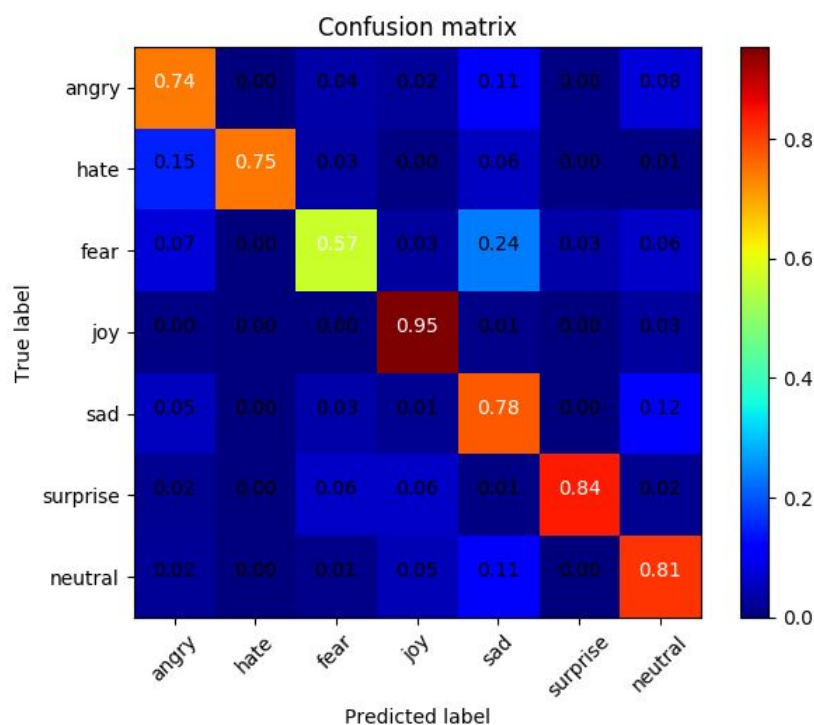
有試著手寫產生DATA的code但發現不會比keras的好，所以果斷放棄

ImageDataGenerator(rotation\_range=20, width\_shift\_range=0.2,  
height\_shift\_range=0.2, rescale=1./255, shear\_range=0.2,  
zoom\_range=0.2, horizontal\_flip=True, fill\_mode='nearest')

	實作前	實作後
acc	0.64864	0.68904

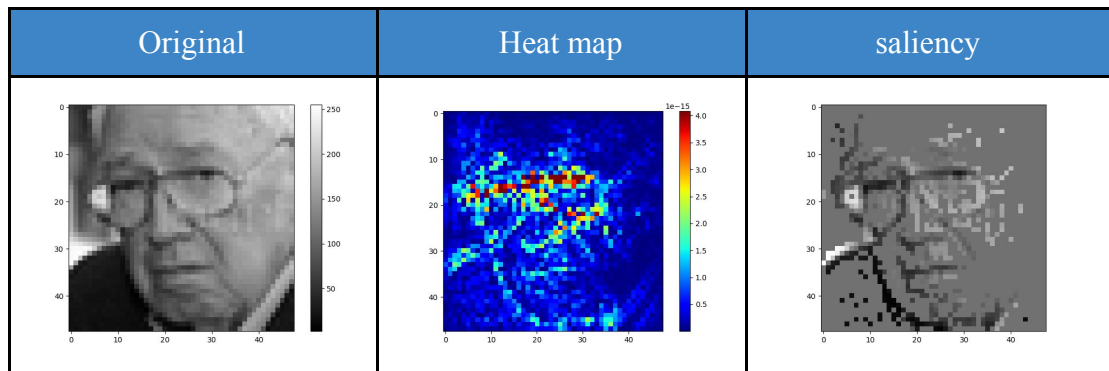
這兩種作法對準確率都有顯著的幫助，在訓練時若是沒有使用data augmentation 則會在training正確率拿到8,90%的正確率validation正確率卻很低6X%的情況，normalization 則可以有效的提高正確率，我猜是因為relu的對於有正規化的資料有比較好的資訊保留，並且減掉mean之後算是把資料的分佈移到中間(標準化)再開始做的意思。

3. (1%) 觀察答錯的圖片中，哪些 class 彼此間容易用混？



可以看到在我的model裡fear會搞混成sad的機率最大hate到angry次之，但其實都不大。

4. (1%) 從(1)(2)可以發現，使用 CNN 的確有些好處，試繪出其 saliency maps，觀察模型在做 classification 時，是 focus 在圖片的哪些部份？



很明顯得出的結果還是著重在辨識眼睛與嘴巴還有些許眉毛的部位，還有一些臉頰上部，還有因為是sad所以法令紋也是個很重要的feature.

5. (1%) 承(1)(2)，利用上課所提到的 gradient ascent 方法，觀察特定層的filter最容易被哪種圖片 activate。

