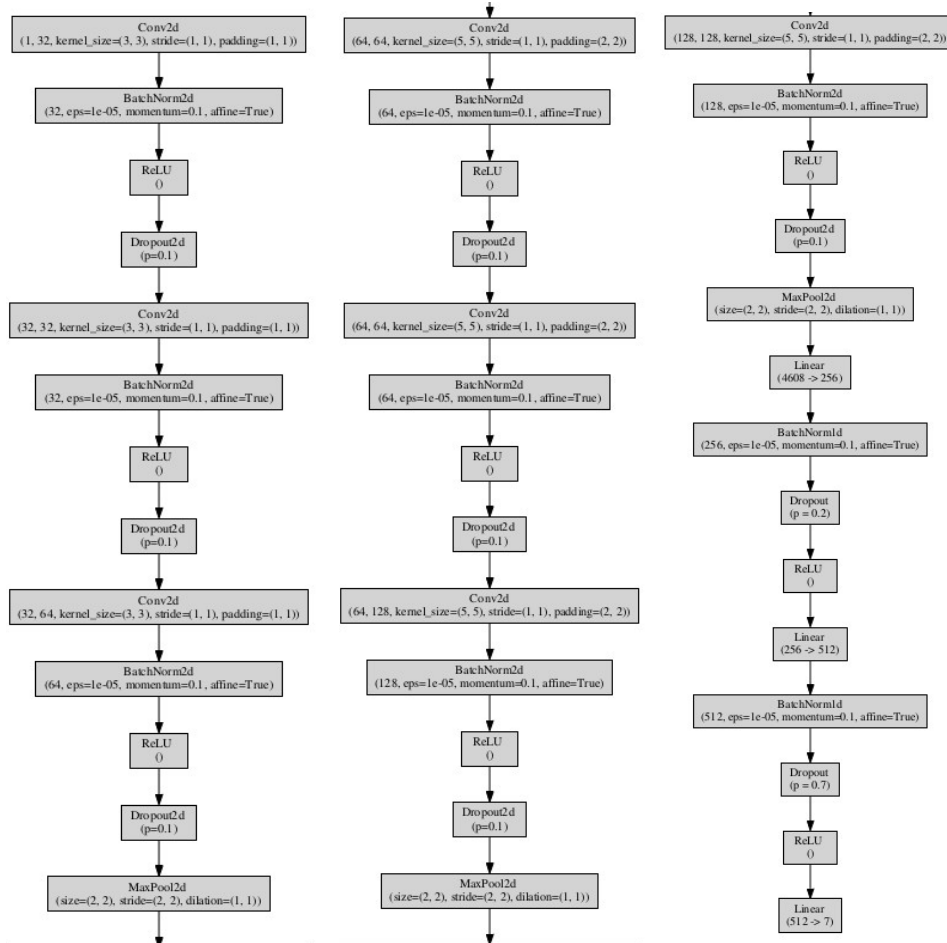


1. (1%) 請說明你實作的 CNN model，其模型架構、訓練過程和準確率為何？
(Collaborators:)

答：CNN model 架構如下：



在 training 部分, 約 30 epoch 左右, Batch size = 128. 使用 Adam Optimizer, learning rate 0.002 (前 10 epoch learning rate = 0.01). 另外在每一個 epoch, 亂數 shuffle data 一次, 並在每個 epoch 有 50% 機率 鏡像 training data (上下 or 左右) 作為 data argumentation. 在 validation 以及 Kaggle 上達到約 63% accuracy.

2. (1%) 承上題，請用與上述 CNN 接近的參數量，實做簡單的 DNN model。其模型架構、訓練過程和準確率為何？試與上題結果做比較，並說明你觀察到了什麼？
(Collaborators:)

答：在我的 CNN model 上, 約有 1M 左右的參數量. 實作的 simple DNN model 架構如下:

```

(Dnn): Sequential (
  (0): Linear (2304 -> 64)
  (1): BatchNorm1d(64, eps=1e-05, momentum=0.1, affine=True)
  (2): ReLU ( )
  (3): Dropout (p = 0.1)
  (4): Linear (64 -> 64)
  (5): BatchNorm1d(64, eps=1e-05, momentum=0.1, affine=True)
  (6): ReLU ( )
  (7): Dropout (p = 0.2)
  (8): Linear (64 -> 64)
  (9): BatchNorm1d(64, eps=1e-05, momentum=0.1, affine=True)
  (10): ReLU ( )
  (11): Dropout (p = 0.7)
  (12): Linear (64 -> 7)
)

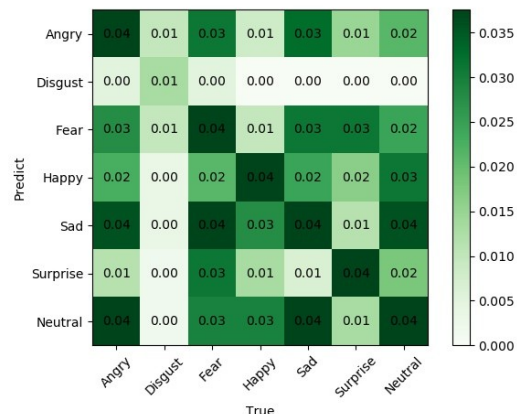
```

同樣使用 Batch size = 128, Adam Optimizer, learning rate 0.002. traning with 35 epoch. 在每一個 epoch , 亂數 shuffle data 一次, 並在每個 epoch 有 50% 機率 鏡像 training data (上下 or 左右) 作為 data argumentation. 其 validation 準確率為 39.5% . 其 performance 比 CNN 來的差很多.

3. (1%) 觀察答錯的圖片中, 哪些 class 彼此間容易用混? [繪出 confusion matrix 分析]
(Collaborators:)

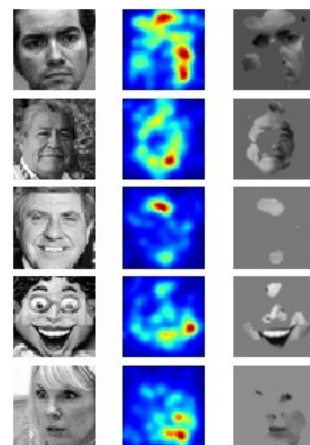
答 :

從觀察, 發現 Sad 與 angry, Fear 和 Neutral (負面情緒之間) 最容易弄混



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?
(Collaborators: 顏敬哲)

答 : 從 saliency maps 發現大多數其 focus 在嘴巴 (嘴角) 以及眼睛.



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。
(Collaborators: 顏敬哲)

答 :

左圖為第 16 層 過 ReLU 後的 filter output, 可以發現某些 layer 容易被具有垂直或水平 feature 的圖片 activate

