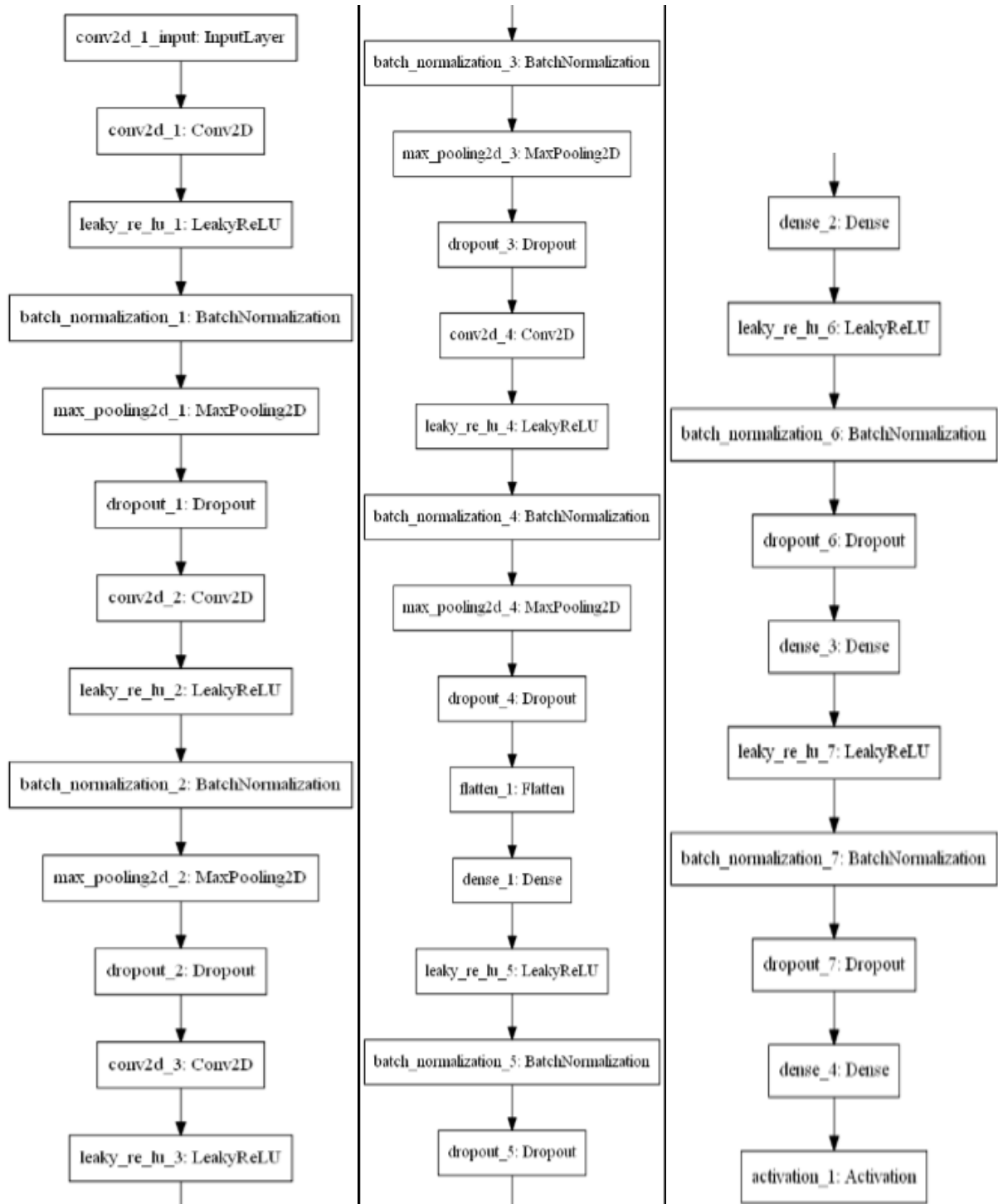


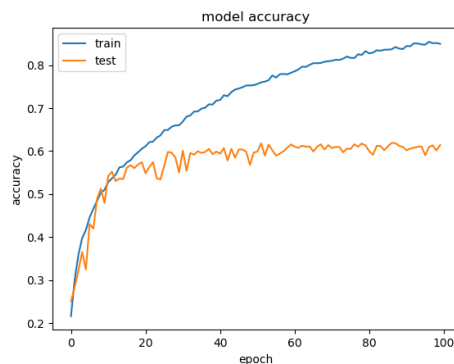
1. (1%) 請說明你實作的 CNN model，其模型架構、訓練過程和準確率為何？

(Collaborators:王垣尹)

答：



Input data 之後先做 4 層 CNN，node 數為 64、128、256、512，只有第一層是 5*5，其餘 3 層為 3*3，flatten 之後再接 4 層 DNN 做 output，node 數為 512、256、128、7，作為 output，每一層都有加上 dropout，CNN 部分從 0.2~0.5，DNN 都是 0.5。總共 train 100 個 epoch，大概到 40 個 epoch 之後就不太會上升了，最後準確率大約是 60%，總參數約 400 萬個



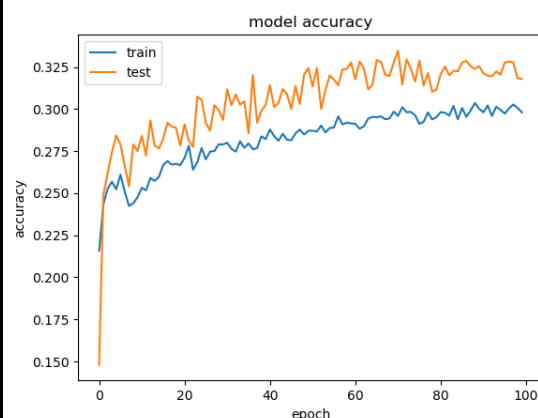
2. (1%) 承上題，請用與上述 CNN 接近的參數量，實做簡單的 DNN model。其模型架構、訓練過程和準確率為何？試與上題結果做比較，並說明你觀察到了什麼？

(Collaborators:NO)

答：

做 5 層 DNN model，每層 1024 個 node 除了 output layer，跟 CNN 比起來光是 train 的部分就作不起來了，最後準確率只有 30%左右

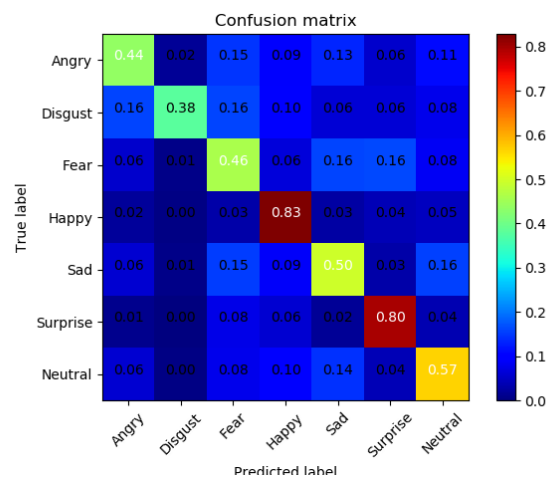
| layer (type) | Output Shape | Param # |
|---|-------------------|---------|
| max_pooling2d_1 (MaxPooling2) | (None, 24, 24, 1) | 0 |
| dropout_1 (Dropout) | (None, 24, 24, 1) | 0 |
| flatten_1 (Flatten) | (None, 576) | 0 |
| dense_1 (Dense) | (None, 1024) | 590848 |
| leaky_re_lu_1 (LeakyReLU) | (None, 1024) | 0 |
| batch_normalization_1 (Batch Normalization) | (None, 1024) | 4096 |
| dropout_2 (Dropout) | (None, 1024) | 0 |
| dense_2 (Dense) | (None, 1024) | 1049600 |
| leaky_re_lu_2 (LeakyReLU) | (None, 1024) | 0 |
| batch_normalization_2 (Batch Normalization) | (None, 1024) | 4096 |
| dropout_3 (Dropout) | (None, 1024) | 0 |
| dense_3 (Dense) | (None, 1024) | 1049600 |
| leaky_re_lu_3 (LeakyReLU) | (None, 1024) | 0 |
| batch_normalization_3 (Batch Normalization) | (None, 1024) | 4096 |
| dropout_4 (Dropout) | (None, 1024) | 0 |
| dense_4 (Dense) | (None, 1024) | 1049600 |
| leaky_re_lu_4 (LeakyReLU) | (None, 1024) | 0 |
| batch_normalization_4 (Batch Normalization) | (None, 1024) | 4096 |
| dropout_5 (Dropout) | (None, 1024) | 0 |
| dense_5 (Dense) | (None, 7) | 7175 |
| activation_1 (Activation) | (None, 7) | 0 |
| Total params: 3,763,207 | | |
| Trainable params: 3,755,015 | | |
| Non-trainable params: 8,192 | | |



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:NO)

答：sad <-> neutral 這兩個彼此被搞混機率最高



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:)

答：

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:)

答：