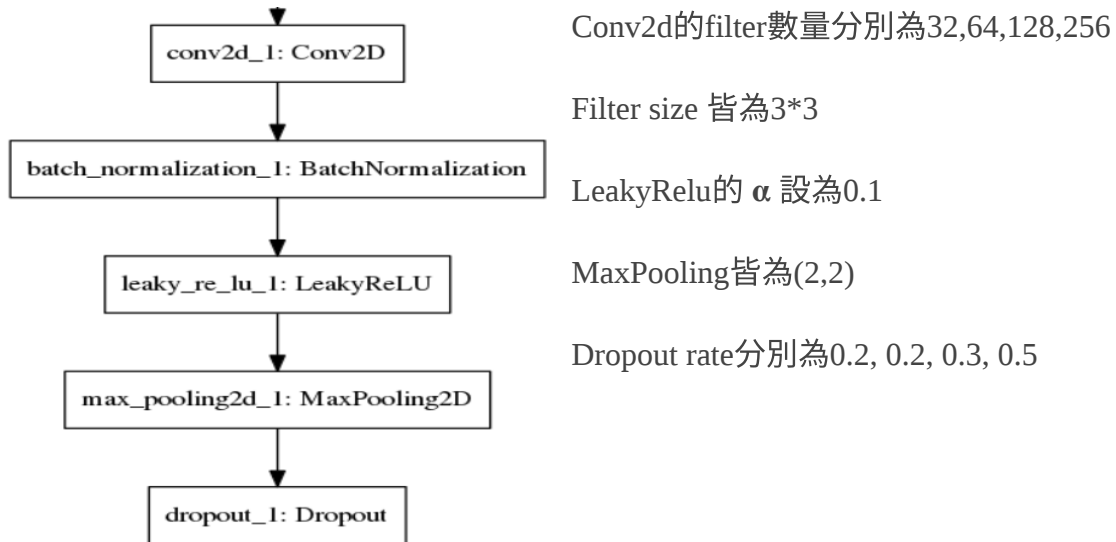


1. (1%) 請說明你實作的 CNN model，其模型架構、訓練過程和準確率為何？  
我大致的架構為：

**Input layer => 四層Conv2d => Dense => Dense(softmax) => Output layer**

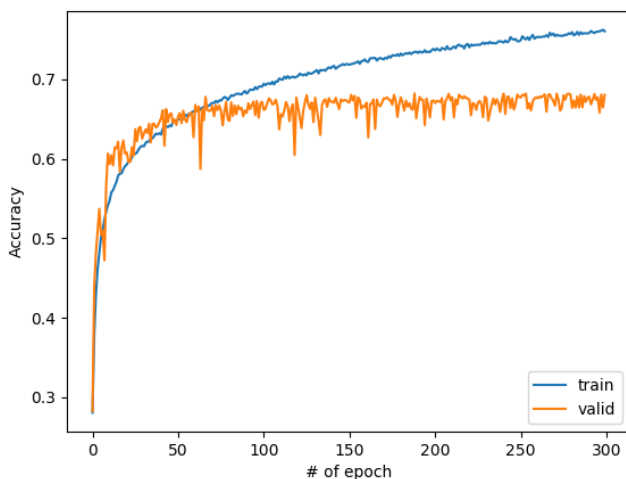
而每個 Convolution layer 詳細架構如下：



最後是Dense(1024) => BatchNormalization => LeakyRelu( $\alpha = 0.1$ ) => Dropout(0.4) => Dense(7, softmax) => output

Total params 為 2,761,351

在資料處理方面，我用Image generator使data有機會稍微旋轉、平移與縮放。



我取了7000筆data當作validation set。

在大約70 epochs時，validation accuracy便開始收斂到約67%的準確率，而training accuracy 則持續升高，且未見收斂趨勢，顯見已overfit。

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？

dense_1 (Dense)	(None, 800)
batch_normalization_1 (Batch Normalization)	(None, 800)
leaky_re_lu_1 (LeakyReLU)	(None, 800)
dropout_1 (Dropout)	(None, 800)
dense_2 (Dense)	(None, 512)
batch_normalization_2 (Batch Normalization)	(None, 512)
leaky_re_lu_2 (LeakyReLU)	(None, 512)
dropout_2 (Dropout)	(None, 512)
dense_3 (Dense)	(None, 256)
batch_normalization_3 (Batch Normalization)	(None, 256)
leaky_re_lu_3 (LeakyReLU)	(None, 256)
dropout_3 (Dropout)	(None, 256)
dense_4 (Dense)	(None, 1024)
batch_normalization_4 (Batch Normalization)	(None, 1024)
leaky_re_lu_4 (LeakyReLU)	(None, 1024)
dropout_4 (Dropout)	(None, 1024)
dense_5 (Dense)	(None, 7)

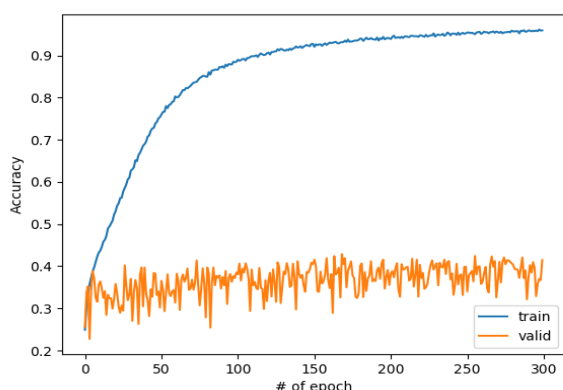
架構：

**Dense(800) => Dense(512) => Dense(256) => Dense(1024) => Dense(7)**

前四層架構皆為Dense => BatchNormalization => LeakyReLU( $\alpha = 0.1$ ) => Dropout

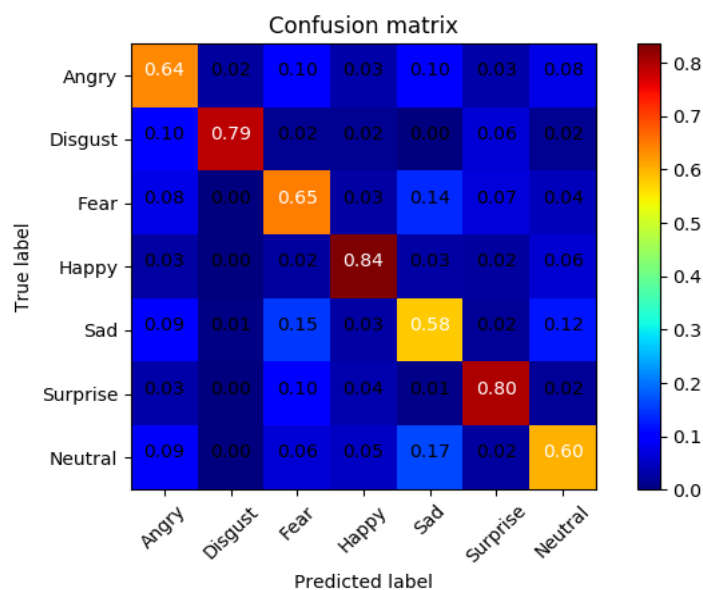
Dropout分別為0.3, 0.4, 0.3, 0.4。

Total params: 2,666,151



雖然參數量差不多，但準確率與CNN差非常多，雖然在大約100 epochs時training accuracy已達九成，但validation accuracy卻只有約四成，推估是因為DNN沒辦法辨認出圖片的pattern，因此就算他將training set fit得很好，但相同表情只要平移或旋轉，只要在圖片上的絕對位置不同，他變無法辨認，因此對validation set的預測非常差。

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]



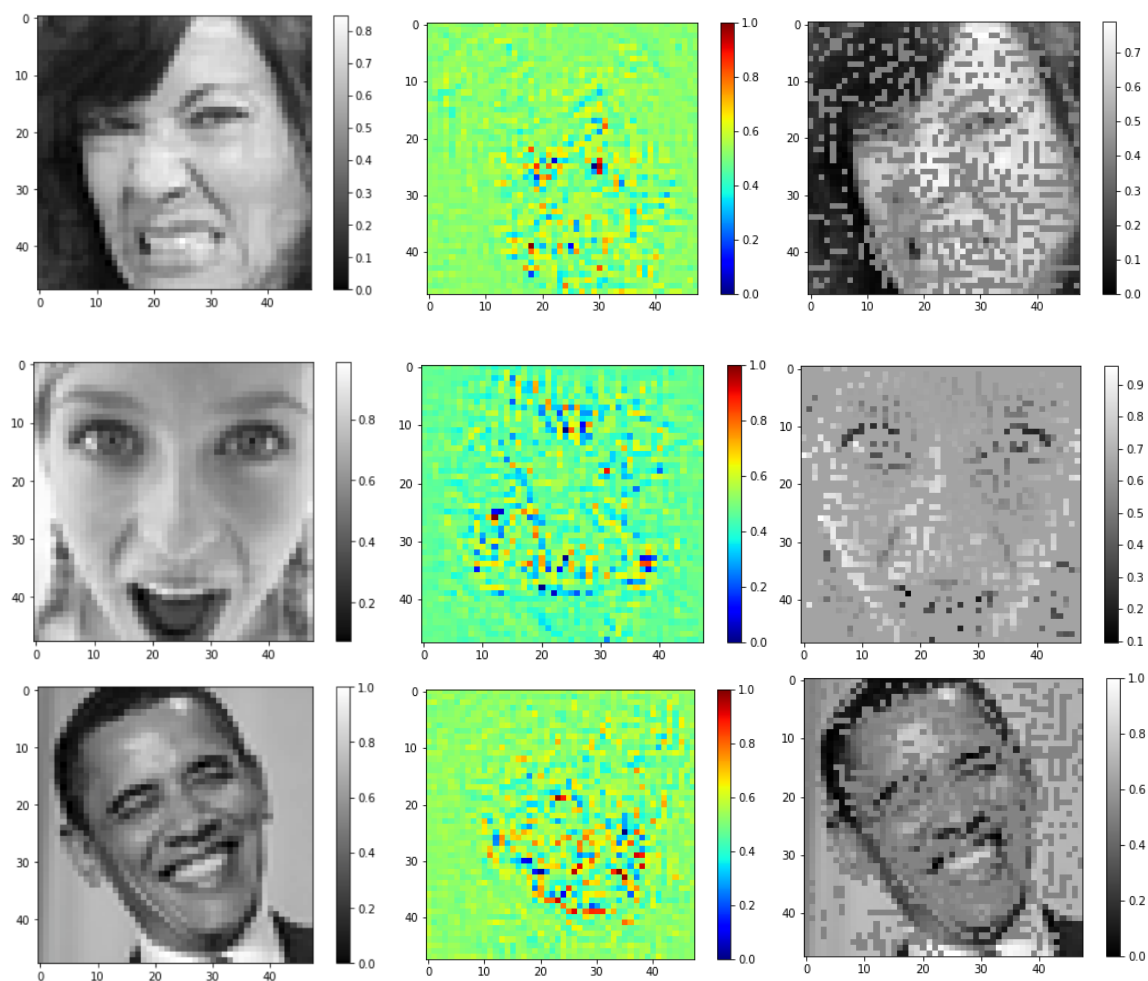
Neutral有17%的機率被認成Sad。

Sad有15%機率被認成Fear。

這三種表情最容易被認錯。

而辨認率最差的則為Sad，只有58%，最好的則為Happy，有84%辨識正確

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？



我舉的例子為厭惡、驚訝或高興的表情，可見他主要是focus在嘴巴的部份，而嘴巴也是我們判斷表情的主要依據。其次則為臉頰到眼睛的部份，推論是因為嘴巴形狀在表情上有變化時會牽動臉頰的肌肉，故也是個很好的判斷依據。

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。

我取了第二層convolution layer學習出的filter，從中取出32個，推測因為model複雜，每層之間可辨認較細節的特徵，因此看不出較為明顯的五官，但仍可以看出幾個 pattern:

- 細線條，佔大多數，只是轉不同角度而已。推測是在辨認臉上較細小的紋路，如：皺紋。
- 稍微較粗的線條，有4個。推測是辨認臉上較明顯的線條，如：五官。
- 較大面積的紋路，有幾個是整張幾乎都是很淡的顏色，有幾個幾乎整張都是很深的顏色。推測是辨認臉上較大面積、平滑的特徵，如：皮膚、頭髮



而我取了training set中id=10000的data。通過第一層convolution layer後，雖不能focus在特定的五官上，但可見已將臉的輪廓、眼睛、嘴巴filter出來。而圖片通過顏色較深的filter也會輸出較深的圖片。

