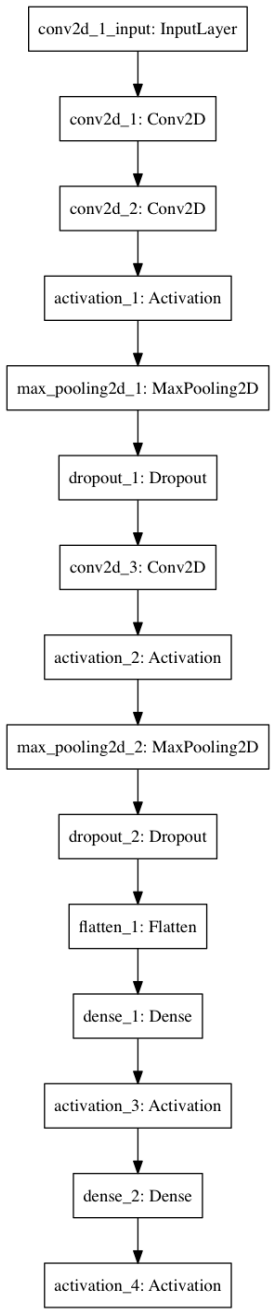
學號：T05902136 系級： 資工一 姓名：Jenny Zhang (張臻凝)

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？
   答：

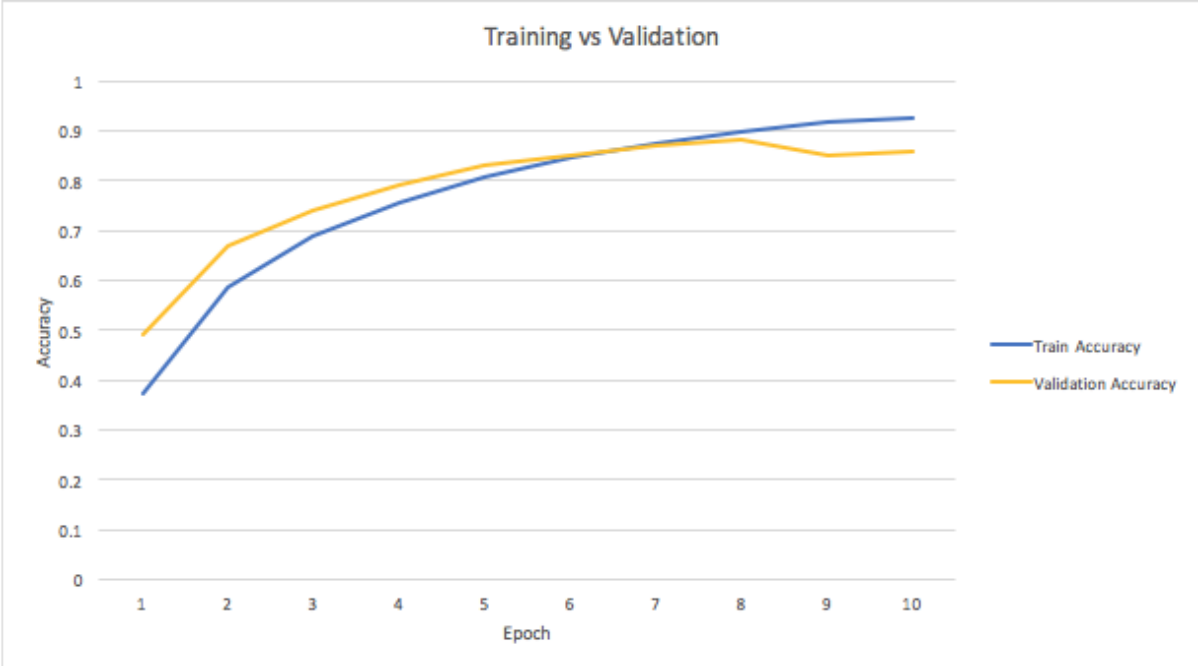


The above diagram is generated from keras.vis_utils.plot_model. Model: Conv2D(32, 3, 3) - Conv2D(32, 3, 3) - Activation('relu') - MaxPooling2D(2, 2) - Dropout(0.25) - Conv2D(64, 3, 3) - Activation('relu') - MaxPooling2D(2, 2) - Dropout(0.25) - Flatten() -

Dense(1028) - Activation(‘relu’) - Dropout(0.25) - Dense(7) - Activation(‘softmax’)

I trained my model with TensorFlow and ran it on CPU. It took about 200-300 seconds per epoch. My batch size was 80 because it seemed to perform the best with this size. Batch size greater than 160 on my model decreased my performance. The pool size and filter size were (2,2) and (3,3), which were common numbers that many people online used. Due to time constraints, I only had time to run my model for 10 epochs.

I had tried adding more convolutional layers and using an image generator to pro-process my images, but neither of them seemed to improve my accuracy using this model. However, by oversampling each classes, which was adding more augmented samples to my data set,my accuracy did improve for about 3%. Without over-sampling, my Kaggle accuracy was 55%. With it, my Kaggle accuracy went up to 58%. From the accuracy graph below, we could see that the accuracy of my validation set stopped growing and went under training accuracy. I think this is because there is an overfitting problem. I had made several modifications to my model to fix this issue but none of them worked. :(Next time, if I have a faster computer, I would try training with different regularizations, dropout rate, DropConnect, and various methods to get a better model.



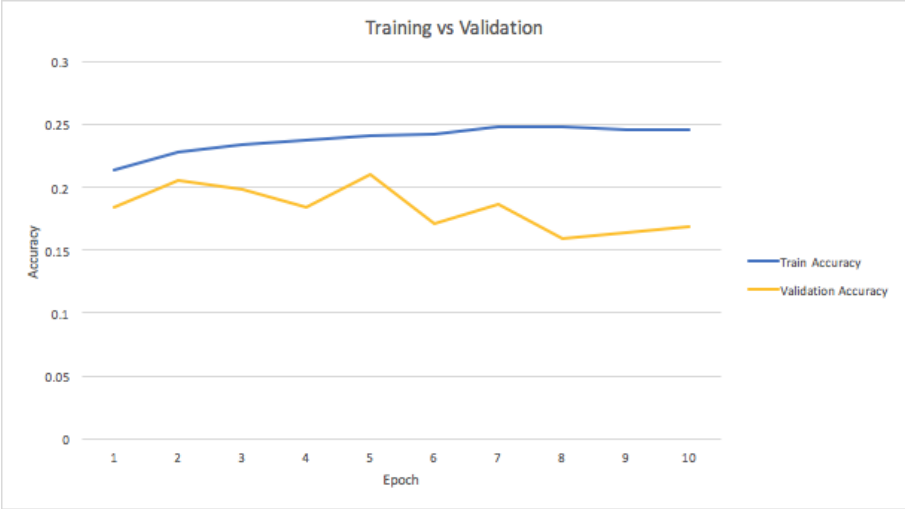
2. (1%) 承上題，請用與上述 CNN 接近的參數量，實做簡單的 DNN model。其模型

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？
答：

My DNN model:

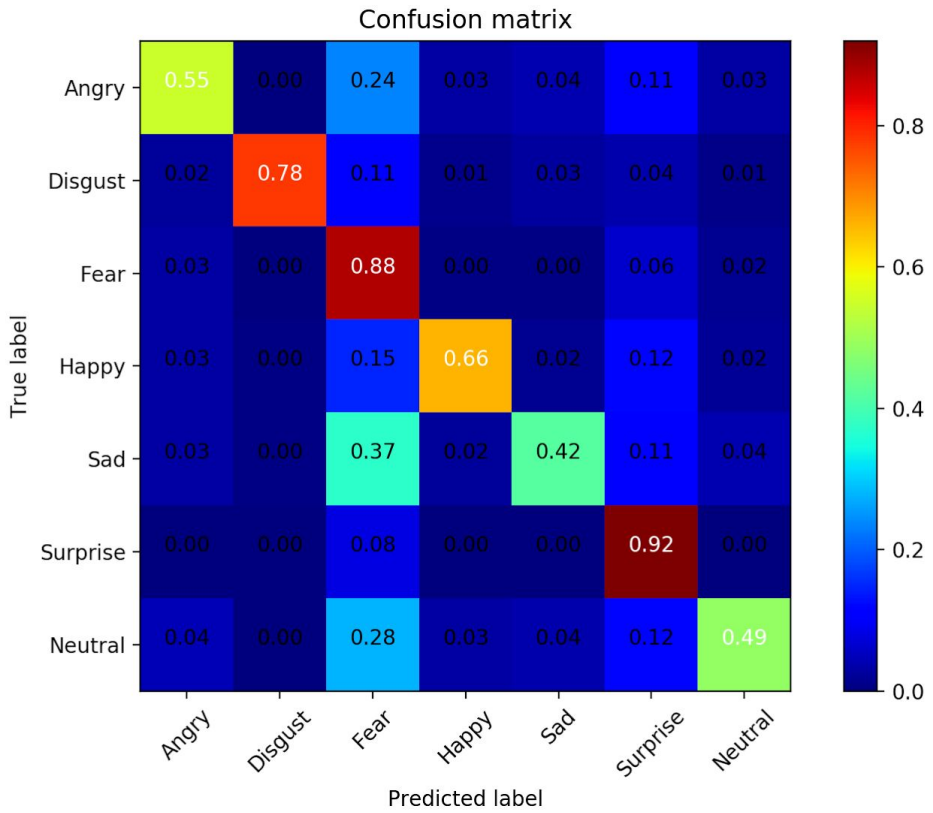
```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
max_pooling2d_1 (MaxPooling2 (None, 24, 24, 1)         0
_________________________________________________________________
dropout_1 (Dropout)          (None, 24, 24, 1)         0
_________________________________________________________________
activation_1 (Activation)    (None, 24, 24, 1)         0
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 12, 12, 1)         0
_________________________________________________________________
dropout_2 (Dropout)          (None, 12, 12, 1)         0
_________________________________________________________________
flatten_1 (Flatten)          (None, 144)               0
_________________________________________________________________
dense_1 (Dense)              (None, 1028)              149060
_________________________________________________________________
activation_2 (Activation)    (None, 1028)              0
_________________________________________________________________
dropout_3 (Dropout)          (None, 1028)              0
_________________________________________________________________
dense_2 (Dense)              (None, 7)                 7203
_________________________________________________________________
activation_3 (Activation)    (None, 7)                 0
=================================================================
Total params: 156,263
Trainable params: 156,263
Non-trainable params: 0
```

This is my DNN model. All parameters were the same except with the convolutional layers removed. From the diagram below, we could see that the accuracy is significantly lower. Furthermore, the validation accuracy started to decrease near the end of epochs, which might indicate an overfitting.



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]

答：



This confusion matrix is generated from the validation set that I used during training. From the matrix, we can see that “Surprise” is predicted correctly 92% (highest) and “Sad” is predicted correctly 42%(lowest). In addition, the model often confuses other emotions with “Fear”. For example, 37% of true label “Sad” was predicted as “Fear”. This confusion matrix result is actually different from what I have expected. This is because there are 7215 samples of “Happy” so I would imagine “Happy” would have the highest accuracy. “Disgust” is the class with least samples but it looks not bad in the matrix. Maybe more image augmentations and data filtering would result in a better model.

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？

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。

答：