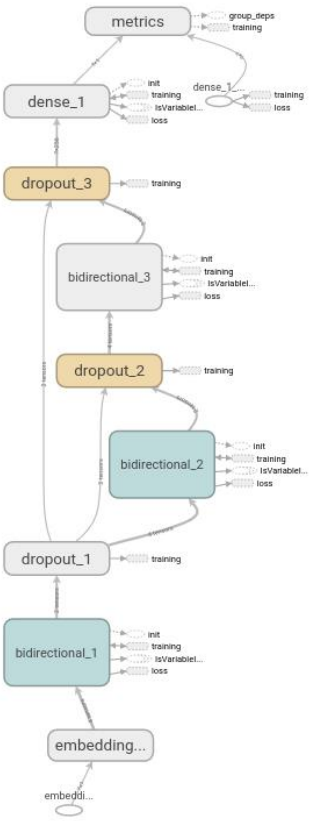
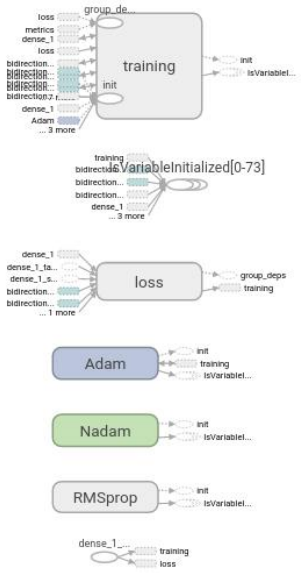
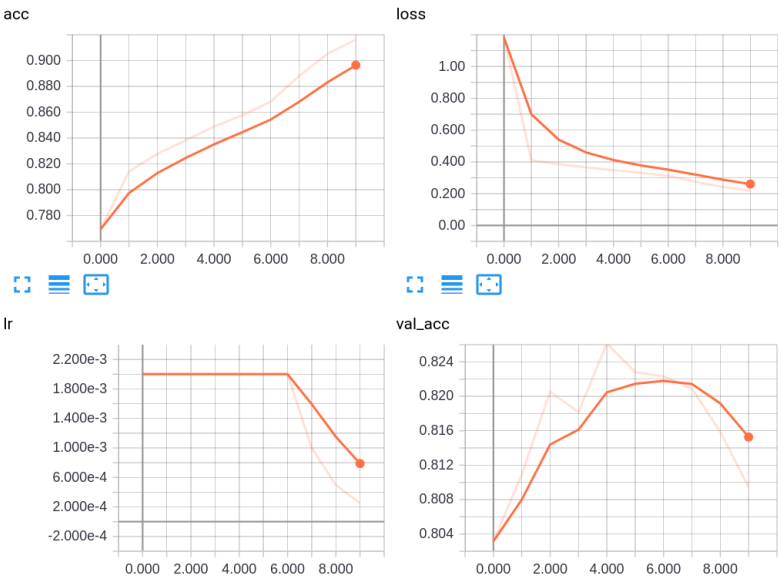
學號：B04705018 系級： 資管三 姓名：楊之郡

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

Word2Vec 使用的是 gensim 生成的 vector，有 512 維，包含 training, testing, no_label data





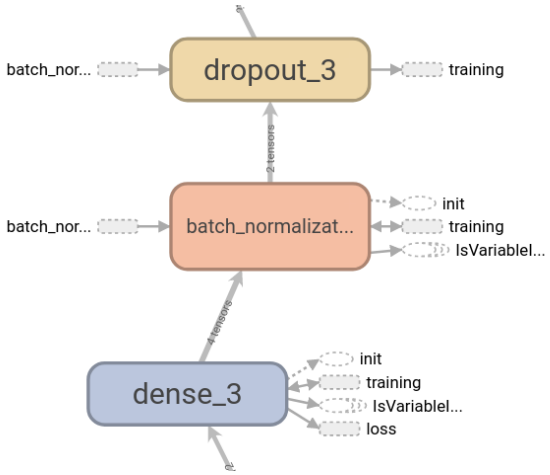


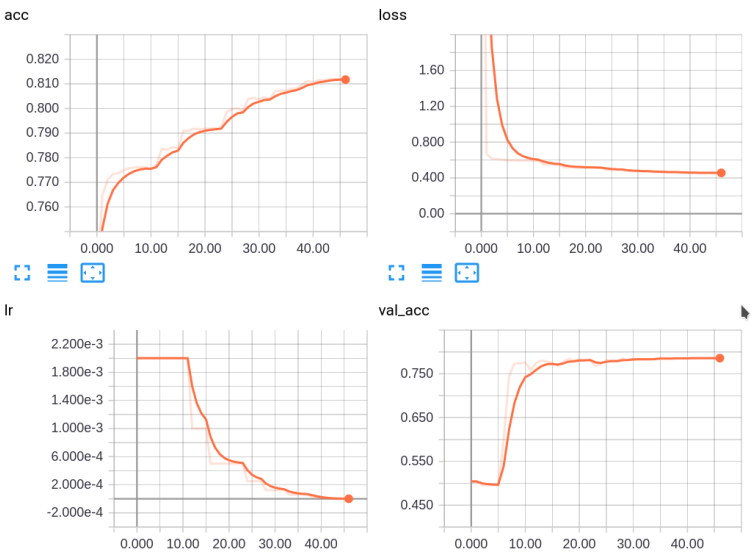
Optimizer 是使用 adam，lr = 0.002，clipvalue = 3，當 val_loss 沒有變小時，將 lr 減半，選擇是第 5 個 epoch 時的 model，因為他 val_loss 最低，最後 Kaggle 上的 accuracy:　public = 0.82704, private = 0.82674

2.　(1%) 請說明你實作的 BOW model，其模型架構、訓練過程和準確率為何？(Collaborators:)

(因為空間不夠，所以貼上部分的 model 的 hidden layer，最後一層為 unit = 1 的 Dense layer)

總共有 6 層 512 unit 的 Dense layer，結構如下，droupout 設 0.2，kernel_regularizer = 0.01，最後一層接 unit = 1 的 dense，所有 activation function 都是 sigmoid，所有 layer 都做 regularization，lr 跟 optimizer 都跟 RNN 一樣，選第 41 個 epoch 的 model





Kaggle accuracy: public:0.78743, private:0.78710

3. (1%) 請比較 bag of word 與 RNN 兩種不同 model 對於"today is a good day, but it is hot"與"today is hot, but it is a good day"這兩句的情緒分數，並討論造成差異的原因。
(Collaborators:)

RNN: y 值 = 0.08190345, 0.99131525

BOW: y 值 = 0.6830564, 0.6830564

這樣的結果應該是因為 RNN 中的 word2vec 會考慮到 word 之間的關聯性，所以算出的 y 值較極端，較能夠分辨 label，而 BOW 因為是利用 word 出現的頻率當 input，所以當兩句話用的 word 一樣，兩個句子的 input vector 其實是長一模一樣的，因此 y 值也會一樣。

4. (1%) 請比較"有無"包含標點符號兩種不同 tokenize 的方式，並討論兩者對準確率的影響。
(Collaborators:)

Public: 0.82082, private: 0.82009，準確率下降的原因應該是標點符號對於句子的正負面還是有以應程度的影響力，像是…可能就是表示無言所以是偏負面，或是!可能表示偏正面。

5. (1%) 請描述在你的 semi-supervised 方法是如何標記 label，並比較有無 semi-surpervised training 對準確率的影響。
(Collaborators:)

將 unlabel 的 data 中 y 值大於 0.82 的 label 設為 1，小於 0.18 的設為 0

將 y 界在 0.82 跟 0.18 中間的 data 都捨棄不用，只使用較 y 值較極端的 data

最後 kaggle 的 public accuracy = 0.82977 比沒做 semi(0.82704)高
Private = 0.82869 也比高於沒做 semi(0.82869)的