

學號：r04921094 系級：電機碩二 姓名：葉孟元

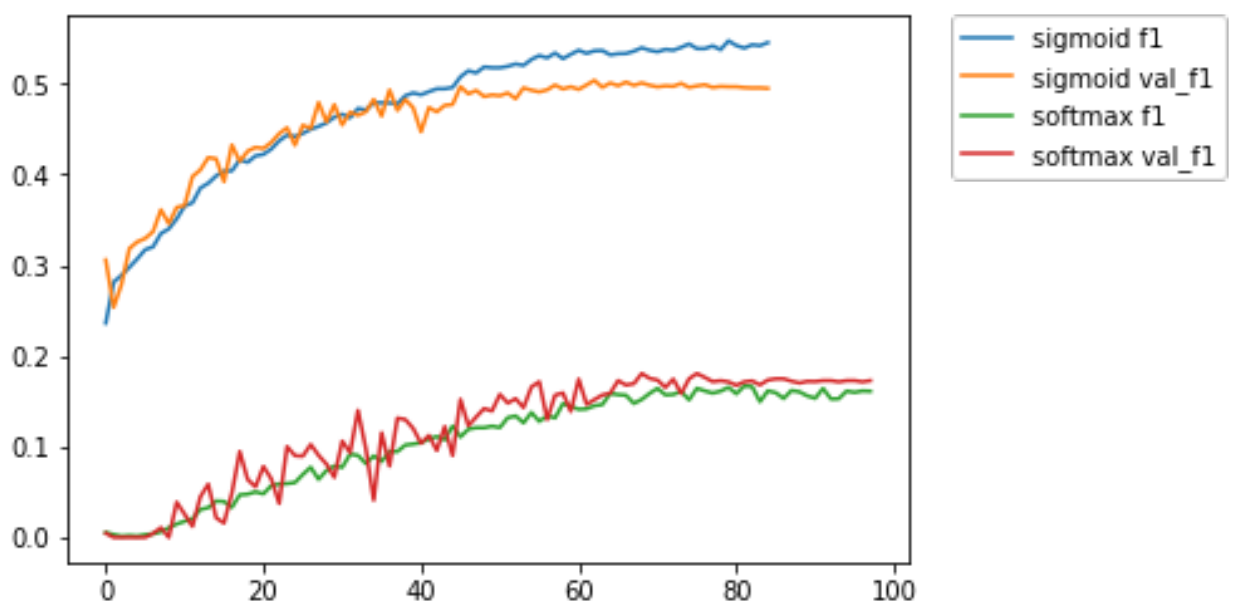
注：因為最後不小心複寫了 tokenizer，所以重新 train 了 ensemble 的模型，public 上比之前好高，但是 private 上並無法和最高分一樣，但是很接近，望助教諒解。

1. (1%)請問 softmax 適不適合作為本次作業的 output layer? 寫出你最後選擇的 output layer 並說明理由。

答：本次的實驗並不適合使用 softmax，因為 softmax 在最後只會選擇一個 label 作為輸出，首先這和我們的 multi-label 問題不同，其次，由於 softmax 的全部值加起來是 1，這就造成了只要一個輸出提高，就會使得所有其他的輸出降低，個人認為 training 上也會出現震盪的問題。可能無法快速 converge。我在實作中使用的 sigmoid activation function，因為這樣對於每一個 label 都相當於一個 binary classification 的問題，只要回答是或者不是就可以了。對於 label 之間並不會相互影響。而且這次的作業用的是 f1\_score，所以，如果只有一個 label 可想而知 true negative 會很多，從而降低 f1\_score。

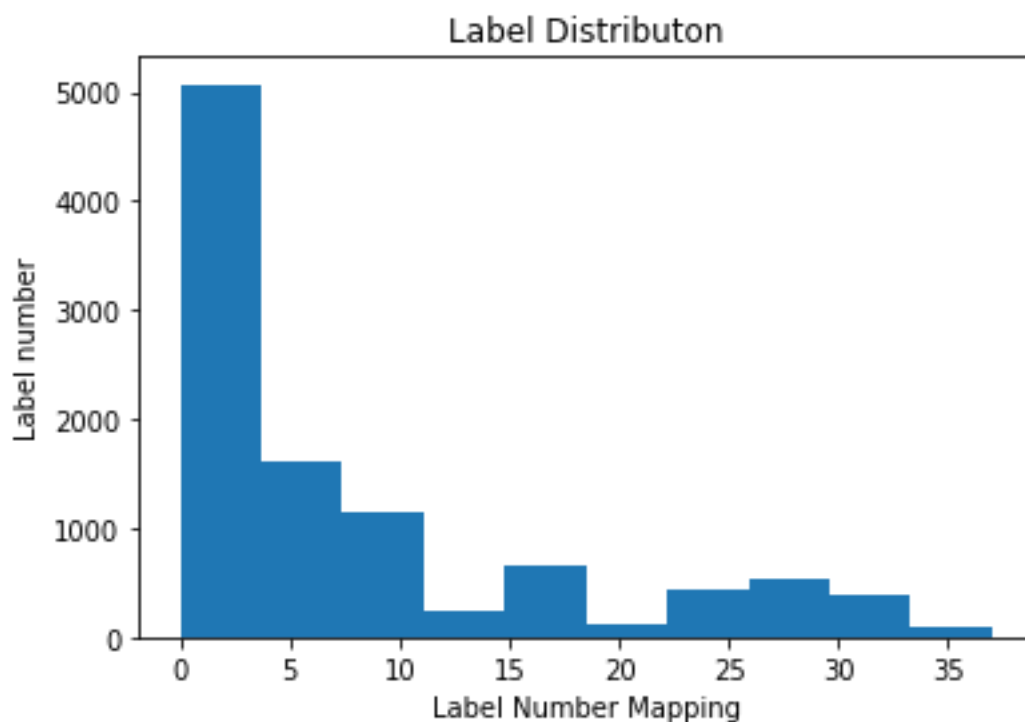
2. (1%)請設計實驗驗證上述推論。

答：實驗模型除了最後一層 Dense 的 activation 不同，分別為 sigmoid 和 softmax，其他的都完全一樣。可以看出 softmax f1 比 sigmoid f1 分數低很多，這就證實了我們的問題是 multi-label 而 softmax 只有一個輸出，並利用 f1\_score 作為衡量標準，導致 f1 很低。因為使用的是 early stopping，softmax 明顯更難 converge 而且在一開始的時候在 validation set 上震盪非常嚴重，也證實了題 1 的推論。



3. (1%)請試著分析 tags 的分布情況(數量)。

答：這次最少的只有 class 只有 11 個，而最多的有 1672 個，雖然想使用 class\_weight 來避免由於某些 class 太少而造成的影響，卻發現在實作中並沒有很好的效果。



4. (1%)本次作業中使用何種方式得到 word embedding?請簡單描述做法。

答：本次使用的 glove 中的 200 維 word embedding，基本作法就是在 keras 中增加一個無法 train 的 embedding layer。然後利用 embedding matrix 將已經 train 過的 word vector 加到 embedding layer 中並和 tokenize 過的 word 做 mapping，然後直接將 tokenize 過的 word 轉換為 word vector。

5. (1%)試比較 bag of word 和 RNN 何者在本次作業中效果較好。

答：當然是 RNN + word embedding 效果比較好， bag of words 很快就 overfitting，並且在 validation 上表現的比較不好。雖然在 DNN 上沒有調整過模型，但是由於 bag of words 並沒有考慮詞序的問題，所以也不會比 word embedding 更好。也是因為這樣而無法使用 RNN，而 RNN 的屬性就是為了解決有重複性和先後順序的 sequence，在我們本次的作業上，RNN 會比較好。以下為實驗結果。

