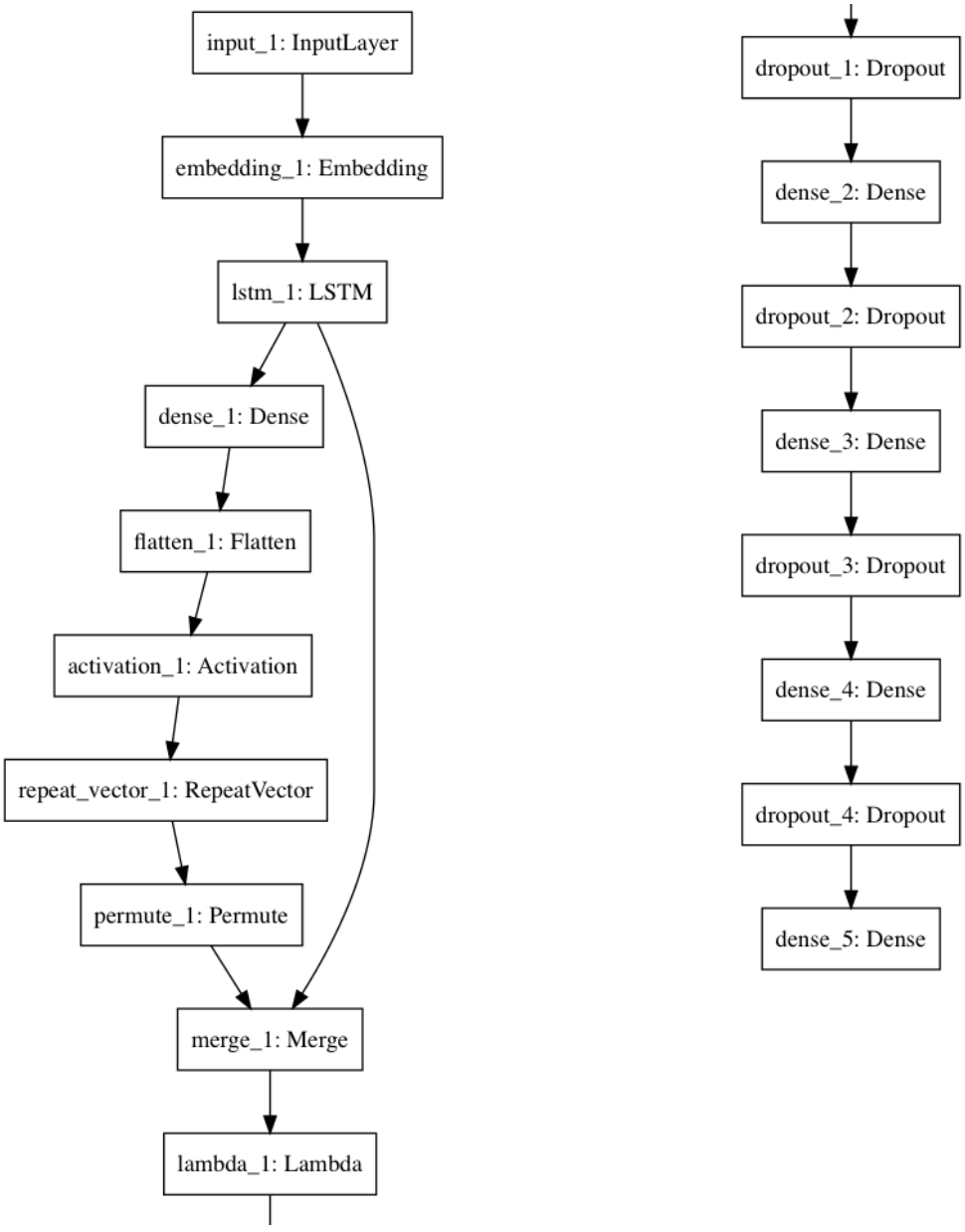
學號：B02902044 系級： 資工四 姓名：陳映紅

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。

- RNN 模型架構圖（因為太長，所以事先裁切）



我在 embedding 和 dense 中間加了 attention layer，然後再接到三個 256 units 的 Dense 層，中間以 Dropout 0.6 連接，最後接到 output layer。

若 output layer 使用 softmax，會讓最後產出結果的機率較集中，對於 multi-class 問題較不利，因此我選用 sigmoid 作為 output layer 的 activation function。

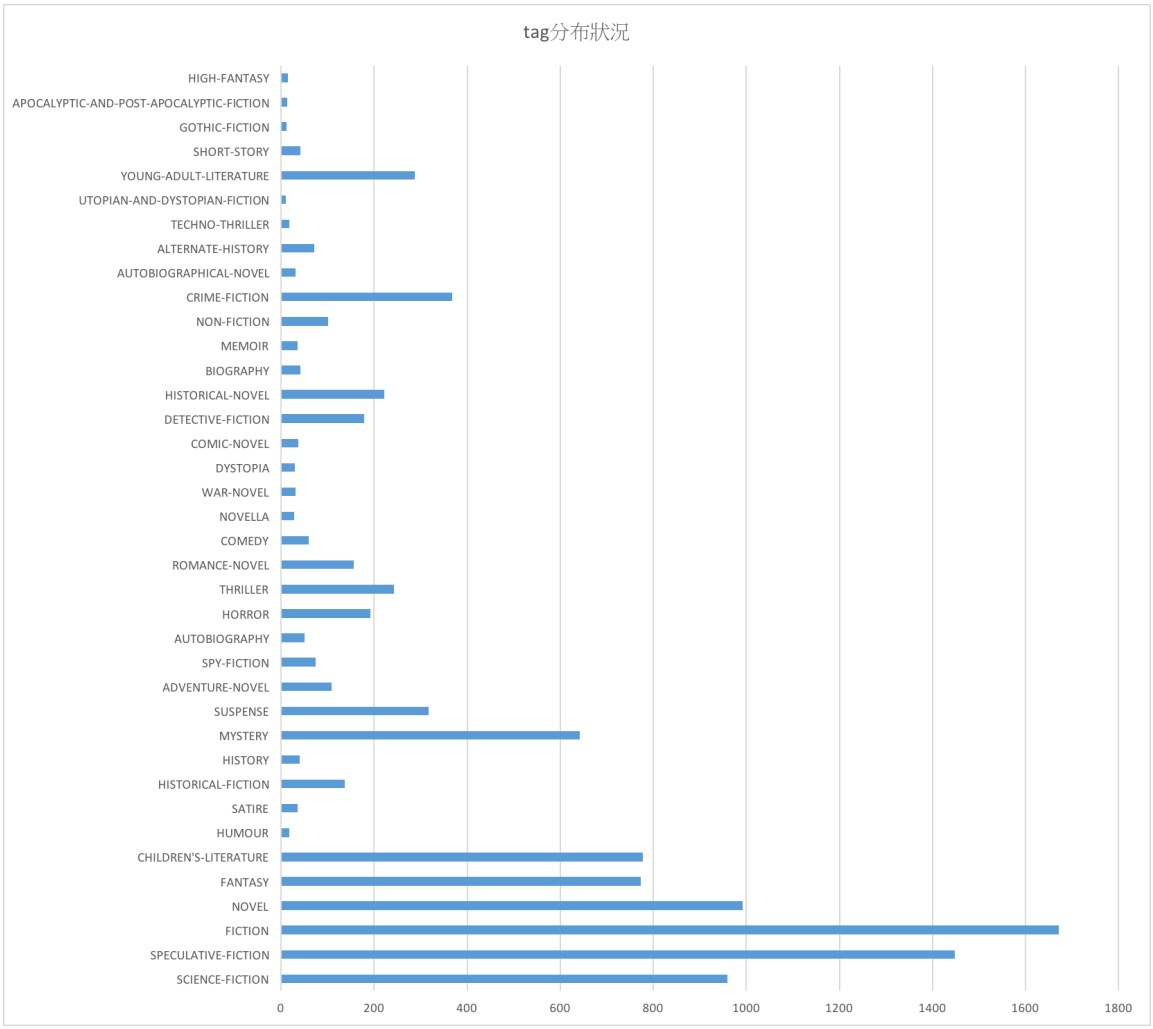
2. (1%)請設計實驗驗證上述推論。

我將上題 RNN 模型的 output layer activation function 改為 softmax，比較改變前後的模型在 test data 上的表現：

Activation	Kaggle public score	Kaggle private score
sigmoid	0.53549	0.51902
softmax	0.17699	0.16143

由上表可知，sigmoid 的表顯明顯優於 softmax。

3. (1%)請試著分析 tags 的分布情況(數量)。



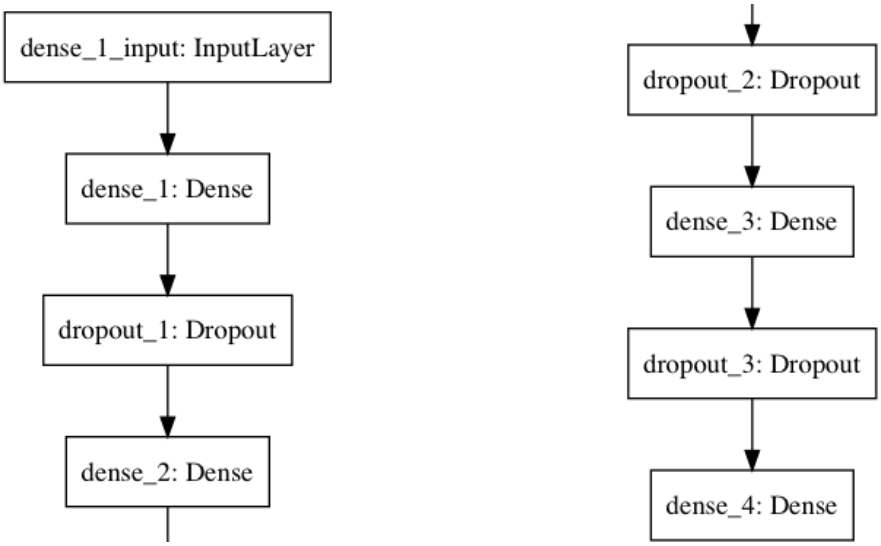
觀察上圖，會發現 fiction 的數量特別多，但也會發現很大一部分的 tag 皆屬於 fiction 的子類別，而 novel 也有同樣的狀況。所以，fiction 和 novel 這兩大類 tag 特別多的原因，應是因為其子類別本來就比較多。

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

我的 RNN 使用 GLOVE pre-trained word vectors，我選用 glove 840B 300 維的 word vector，其使用 Common Crawl 爬蟲蒐集資料後再訓練模型。他們以統計方法算出詞和詞間的 co-occurrence matrix，並以此 matrix 中的非零元素訓練出一個 log-bilinear model。

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

- Bag of word 模型架構（因為太長，所以事先裁切）：



BOW 模型的 Dense 和 Dropout 皆和 RNN 一樣，Dense 有 256 個 unit，Dropout 則為 0.6。

比較 BOW 和 RNN 在 test data 上的表現：

Model	Kaggle public score	Kaggle private score
RNN	0.53549	0.51902
BOW	0.50260	0.49340

若製作 input matrix 時將每個 word 的權重由 binary 改為 tfidf，則表現上升，如下表：

Model	Kaggle public score	Kaggle private score
RNN	0.53549	0.51902
BOW	0.52687	0.52801

原始的 BOW（使用 binary）不考慮每個 word 於每筆 data 中的重要性，影響判斷類別時的準確率，而 RNN 因考慮每個 word 的前後文，因此準確率自然也比 BOW 高。但將 BOW 模型的 binary mode 改為 tfidf 後，每個 word 有不同的權重，機器可以以權重判定哪些 word 出現在特定類別的機率較高，因此準確率也較高，甚至與 RNN 不相上下。