

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。

我的架構是一層 embedding layer、GRU 128(drop 0.25)+tanh、Dense 256(drop 0.1)+relu、Dense 128(drop 0.1)+relu、Dense 64(drop 0.1)+relu、Dense 38+sigmoid，optimizer 是 Adam。我覺得 softmax 不適合這次的作業，因為 softmax 的 output 全部加起來的質是 1，所以最多只會有一個 output 為 1，但是因為這次作業有多個 label，所以理想情況應該要有多個 output 的質都可以為 1，所以我最後選擇用 sigmoid 來當作 output layer，因為每一個 output 的質都可以是 1。

2. (1%)請設計實驗驗證上述推論。

相同的架構下，我用 softmax 當 output layer，很難 train，幾乎是 train 不起來，後來改成用 sigmoid 當 output layer 後 F1 最低都有到 0.4 以上，最好可以到 0.50549。F1 score 是種評估利用 precision 跟 recall 來評估準確度的方法。

3. (1%)請試著分析 **tags** 的分布情況(數量)。

下列是 training data 的 tag 分布情況:

SCIENCE-FICTION	SPECULATIVE-FICTION	FICTION	NOVEL	FANTASY	CHILDREN'S-LITERATURE
959	1448	1672	992	773	777

HUMOUR	SATIRE	HISTORICAL-FICTION	HISTORY	MYSTERY	SUSPENSE
18	35	137	40	642	318

ADVENTURE-NOVEL	SPY-FICTION	AUTOBIOGRAPHY	HORROR	THRILLER	ROMANCE-NOVEL
109	75	51	192	243	157

COMEDY	NOVELLA	WAR-NOVEL	DYSTOPIA	COMIC-NOVEL	DETECTIVE-FICTION
59	29	31	30	37	178

HISTORICAL-NOVEL	BIOGRAPHY	MEMOIR	NON-FICTION	CRIME-FICTION	AUTOBIOGRAPHICAL-NOVEL
222	42	35	102	368	31

ALTERNATE-HISTORY	TECHNO-THRILLER	UTOPIAN-AND-DYSTOPIAN-FICTION	YOUNG-ADULT-LITERATURE	SHORT-STORY	GOTHIC-FICTION
72	18	11	288	41	12

APOCALYPTIC-AND-POST-APOCALYPTIC-FICTION	HIGH-FANTASY
14	15

可以看到前六種小說 SCIENCE-FICTION, SPECULATIVE-FICTION, FICTION, NOVEL, FANTASY, CHILDREN'S-LITERATURE 的 tag 數量最多，相較其他其它就少很多。

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

我是利用 glove 的 dictionary 來得到 word embedding，如果在 glove 裡面找不到的字我就用 dog 這個單字來取代，結果效果還不錯。Glove 是利用 count based 的方式來做，它的原理是去計算任兩個 word 在同一份 document 出現的次數，並且讓兩個單字的 vector 內積後會接近這個次數。

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

我的 bag of word 架構是 Dense 256(drop 0.1)+relu+l2 regu=0.1、4*Dense 256(drop 0.1)+relu、Dense 38+sigmoid，optimizer 是 Adam。Bag of word 的做法相較於 RNN 需要更多的 memory，因為他每個 input 的 size 都太大了，所以它參數也更多，很容易 overfitting，最後做出來的結果 F1 只有 0.31386，而 RNN 可以到 0.50549，所以 RNN 不管是在 memory space 和 accuracy 都比 bag of word 好。