Homework 5 Report - Text Sentiment Classification

學號: R06942018, 姓名: 何適楷, 系級: 電信碩一

1

(1%) 請比較有無 normalize 的差別。並說明如何 normalize.

	public	private
no normalize	0.85279	0.84469
normalize	0.85212	0.86777

我將分數全部都除以 5，所以分數的區間全部都壓在 0-1 中間，結果並不會比較好，一般來說，normalize 的精神應該在於不同 feature 的範圍差異過大，導致更新的比例不同，以這題的情況，應該沒有這種問題，故沒有看到顯著的效果。

2

(1%) 比較不同的 embedding dimension 的結果。

embedding dimension	public	private
32	0.85866	0.85075
64	0.85834	0.84981
128	0.85304	0.84685
256	0.85212	0.84547
512	0.84964	0.84575
1024	0.85171	0.84631

embedding dimension 在 512 的時候最好，之後就開始上升了。

3

(1%) 比較有無 bias 的結果。

	public	private
bias	0.85212	0.84547
no bias	0.85606	0.84669

有 bias 的結果比較好，就如課堂上所言，每種物品或人也可能存在”基本分”之類的量。

4

(1%) 請試著將 movie 的 embedding 用 tsne 降維後，將 movie category 當作 label 來作圖。

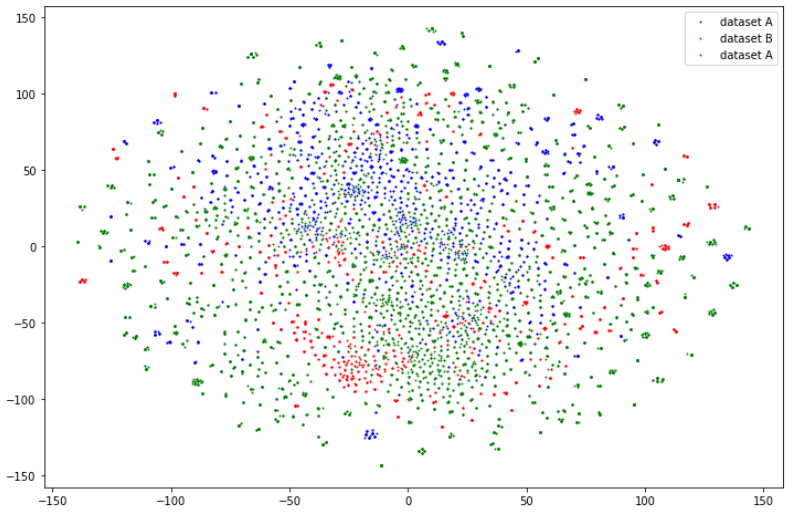

Figure 1: tsne

datasetA = ’ Drama’ ,’ Musical’
datasetB = ’Thriller’,’ Horror’ ,’ Crime’
datasetC = dventure,Animation,Children’s

5

(1%) 試著使用除了 rating 以外的 feature, 並說明你的作法和結果，結果好壞不會影響評分。

我先把已經得到的分數，再與 user 的資訊 (性別、年齡，Occupation) 與 movie 的資訊 (三種類別)，合成一個 7 維的向量，再經過一個單層的 Dense layer，得到一個新的 model。結論是沒有比單純 MF 的好，可能的原因是多餘的 data 幫助並不顯著，反而影響 model 的判斷，或是 model structure 設計的不夠好。model 的 summary 在下頁。

	public	private
multi data model	0.86402	0.85635

ref:https://nipunbatra.github.io/blog/2017/recommend-keras.html

Layer (type)	Output Shape	Param #	Connected to
movie_input (InputLayer)	(None, 1)	0	
user (InputLayer)	(None, 1)	0	
movie_embedding (Embedding)	(None, 1, 512)	2023936	movie_input[0][0]
user_embedding (Embedding)	(None, 1, 512)	3092992	user[0][0]
movie_vec (Flatten)	(None, 512)	0	movie_embedding[0][0]
user_vec (Flatten)	(None, 512)	0	user_embedding[0][0]
embedding_9 (Embedding)	(None, 1, 1)	3953	movie_input[0][0]
embedding_10 (Embedding)	(None, 1, 1)	6041	user[0][0]
dot_5 (Dot)	(None, 1)	0	movie_vec[0][0] user_vec[0][0]
movie_vec_bias (Flatten)	(None, 1)	0	embedding_9[0][0]
user_vec_bias (Flatten)	(None, 1)	0	embedding_10[0][0]
add_5 (Add)	(None, 1)	0	dot_5[0][0] movie_vec_bias[0][0] user_vec_bias[0][0]

Total params: 5,126,922
Trainable params: 5,126,922
Non-trainable params: 0

Figure 2: tsne