Homework 6 Report -Text Sentiment Classifation

學號：B04501095 系級： 土木三 姓名：黃平瑋

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

normalization的方法是將train data 85萬筆的rating資料做標準化的處理, 並將標準差和平均值先儲存起來, 在predict時在套用到testing set上面

	training rmse	public rmse	testing rmse
有normalize	0.6933	0.85989	0.85338
無normalize	0.6157	0.85637	0.85432

(training rmse 因為有除以標準差, 所以比testing資料低了些, 資料的標準差大約為1.1)
由上表可以看出, 我的model在經過標準化的處理後, 其實差異沒有太大

2. (1 %)比較不同的embedding dimension的結果。

dimension	training rmse	public rmse	testing rmse
16	0.6523	0.87063	0.86424
64	0.6833	**0.85989**	**0.85338**
128	0.6725	0.85894	0.85345
256	0.6659	0.86200	0.85626

(training rmse 因為有除以標準差, 所以比testing資料低了些, 資料的標準差大約為1.1)
由上表可以看出不同的embedding dimension其實差距沒有太大, 擁有相近的誤差, 唯一比較大的差距是較高維的latent vector需要較多的epoch才能收斂

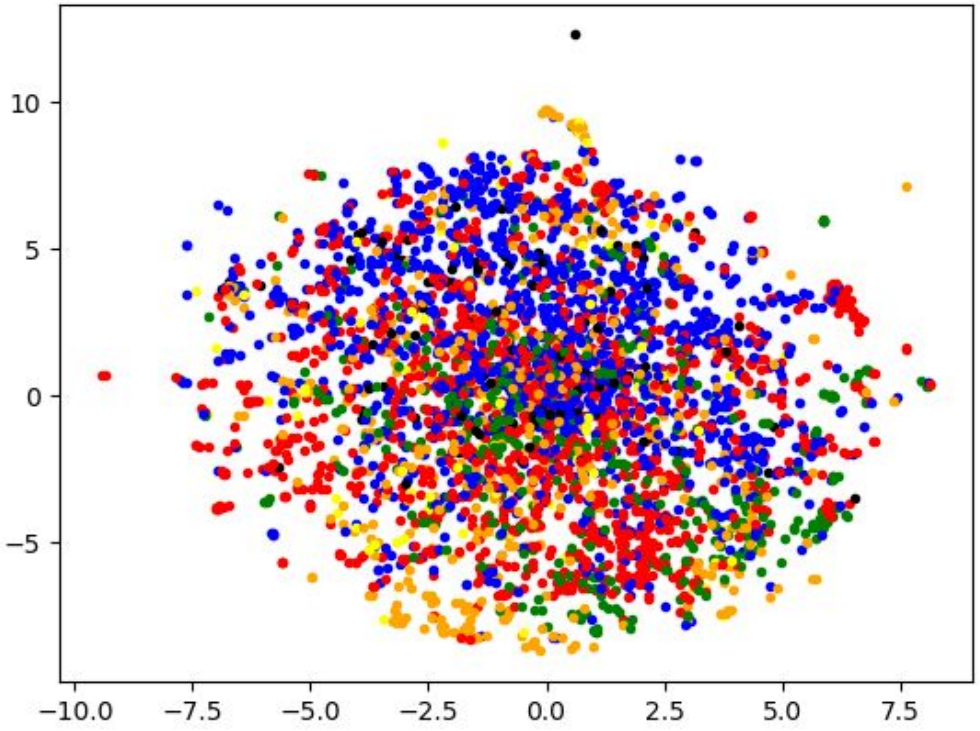
3. (1 %)比較有無bias的結果。

	training rmse	public rmse	testing rmse
有bias	0.6833	**0.85989**	**0.85338**
無bias	0.5256	0.86365	0.85668

(training rmse 因為有除以標準差, 所以比testing資料低了些, 資料的標準差大約為1.1)
在我的實做的model中有加入bias後的表現比沒有加bias的結果稍微好了一點, 但兩者的差異並不大, 而有加bias的需要較少的epoch即可收斂

4. (1 %)請試著將movie的embedding用tsne降維後，將movie category當作label來作圖。

我將幾個類別相近的電影劃分成一群, 總共有六群電影的類別, 取出model input layer 和 movie 的embedding layer將3800筆電影embed成64維的向量, 再使用tSNE降維到二維平面, 結果如下圖：



color	genre
red	Animation, Children's, Comedy
green	Adventure, Action, Western
blue	Romance, Drama, Musical
black	Documentary, War
yellow	Fantasy, Sci-Fi
orange	Crime, thriller, Horror, Mystery, Film-Noir

可以看出不同群的data幾乎是沒有分開的, 推測會有這樣的結果, 是因為model在訓練的target是rating, 所以embedding layer會去maxmize不同電影和**rate**之前的關係 所以就沒有有效的涵蓋到不同種類電影所隱含的feature, 造成最後cluster的結果欠佳

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。

除了使用rating做matrix factorization預測外, 我還加入了其他的feature如age, gender, occupation, genre, 以下是各個feature的處理方法

- **genre**透過one-hot encoding 轉換成18維的向量, 再透過一層dense layer轉換成和embedding dimension相同的維度
- **age**每隔10歲當作一個區間, 大於60歲算一個區間, 透過one-hot encoding可以變成一個7維的向量, 再透過一層dense layer轉換成scalar
- **gender**為0或1的scalar
- **occupation**我也是利用one hot encoding將其劃分成8個區間, 再透過一層dense layer轉換成和embedding dimension相同的維度

將user_id, movie_id, genre, occupation四個向量兩兩彼此做內積, 最後在通過兩層Dense layer

dimension	training rmse	public rmse	testing rmse
64	0.6523	0.88537	0.89284
128	0.6833	0.90061	0.90677
256	0.6725	0.90119	0.90691

這是最後DNN在不同維度上的結果, 可以看到在幾個維度中都沒有比matrix factorization的結果還要好, 可能是network參數過多使得model overfit