

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

	private	public	Val_rmse	Train_rmse
Normalize	0.85014	0.850008	0.8457	0.7773
No Normalize	0.86318	0.86028	0.8602	0.7938

Normalize 的方法，是計算出 Train rating 的平均數和變異數後，先減去平均數再除以變異數，作為 `normalize` 的方法。而從上表，有 `normalize` 的 validation 和是 test 較沒有 `normalize` 的減少 0.01 的 rmse。

2. (1%)比較不同的 latent dimension 的結果。  
(collaborator:)

Latent size	private	public	Val_rmse	Train_rmse
10	0.85014	0.850008	0.8457	0.7773
100	0.85980	0.86034	0.8534	0.7371
500	0.85860	0.85922	0.8550	0.6358

從上表格來看，當 latent size 越小的時候，validation 和 test 的 rmse 都較小，差異都落在 0.01 左右，而 train 的 rmse 則是隨著 latent size 越大表現較好。

3. (1%)比較有無 bias 的結果。  
(collaborator:)

	private	public	Val_rmse	Train_rmse
Bias	0.85014	0.850008	0.8457	0.7773
No Bias	0.85951	0.85817	0.8566	0.7911

從上表格，有 bias 的 rmse 在 test 和 validation 都比沒有 bias 的少 0.01 左右。

4. (1%)請試著用 DNN 來解決這個問題，並且說明實做的方法(方法不限)。  
並比較 MF 和 NN 的結果，討論結果的差異。

(collaborator:)

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	(None, 1)	0	
input_2 (InputLayer)	(None, 1)	0	
embedding_1 (Embedding)	(None, 1, 300)	1812000	input_1[0][0]
embedding_2 (Embedding)	(None, 1, 300)	1185600	input_2[0][0]
flatten_1 (Flatten)	(None, 300)	0	embedding_1[0][0]
flatten_2 (Flatten)	(None, 300)	0	embedding_2[0][0]
concatenate_1 (Concatenate)	(None, 600)	0	flatten_1[0][0] flatten_2[0][0]
dropout_1 (Dropout)	(None, 600)	0	concatenate_1[0][0]
dense_1 (Dense)	(None, 300)	180300	dropout_1[0][0]
dropout_2 (Dropout)	(None, 300)	0	dense_1[0][0]
dense_2 (Dense)	(None, 1)	301	dropout_2[0][0]
Total params: 3,178,201			
Trainable params: 3,178,201			
Non-trainable params: 0			
None			

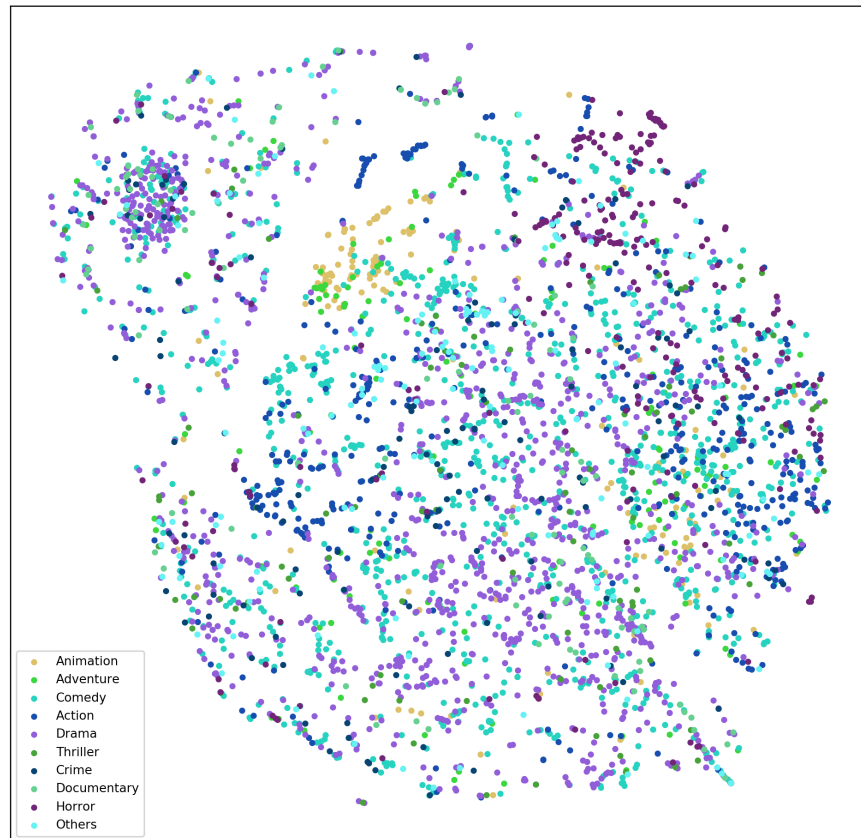
將 user embedding , movie embedding concatenate 起來，並再疊一上層數與其 latent size 相同的 Dense，最後再疊上一層的 Dense。

	private	public	Val_rmse	Train_rmse
MF	0.85014	0.850008	0.8457	0.7773
DNN	0.85733	0.85714	0.8528	0.8193

Dnn 的 rmse 大約比 MF 多個 0.7，效果並沒有很差。但在實作時，如果 DNN 做 normalize 的效果一直都很差，大概都在 1.0 左右，反觀 MF 做 normalize 的效果反而比較好。

5. (1%)請試著將 movie 的 embedding 用 tsne 降維後，將 movie category 當作 label 來作圖。

(collaborator:)



# Children 併入 Animation

# Romance 、Fantasy 併入 Comedy

# Sci-Fi 、Western 、Mystery 、Film-Noir 、War 併成 Others

從圖中，Horror（酒紅）的部分聚集在左上方，Animation（黃色）聚集於中間上半部。而 Drama, Comedy 則是平均分散到全部地方。而其他種類則是分佈不止於一個地方，分別聚集於不同的地方。

6. (BONUS)(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。

(collaborator:)