Homework 6 Report

學號：r06521605
系級：土木所電輔組碩一
姓名：許舜翔

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

   答：

   將原先模型中 Output 的 rating，以 0–1 標準化及 z–score 標準化，並與原先模型精度做比較，結果如下表。

   （使用的模型為 embedding dimension=32，且僅考慮 id 跟 rating）

if normalize	Public score	Private score
0–1 標準化	0.9260	0.9234
z–score 標準化	0.8653	0.8653
無	0.8516	0.8437

   可以發現結果並無改善，甚至變差，但收斂速度有略為提升，推測應為原先 rating 的分佈應無偏差過大的值，所以並沒有無法收斂的情形，normalize 對於提升精度則無幫助。

2. (1 %)比較不同的 embedding dimension 的結果。

   答：

Embedding dimension	Public score	Private score
16	0.8530	0.8447
32	**0.8516**	**0.8437**
64	0.8537	0.8455

128	0.8890	0.8832

結果顯示當維度提升時，對 loss 的改善效果不大，甚至可能因維度過高，而造成 overfitting，反而使得 loss 提高

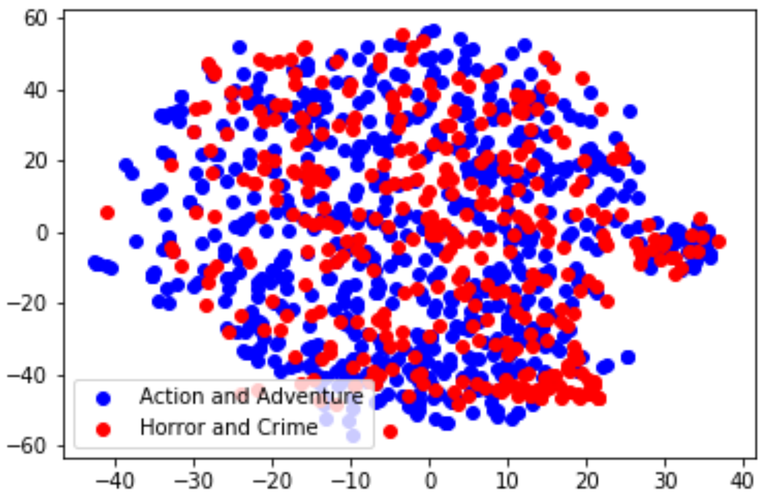
3. (1 %)比較有無 bias 的結果。
答：

模型以 embedding dimension=32，僅考慮 id 跟 rating 的情況下進行比較，結果如下表。

if consider bias	Public score	Private score
有	**0.8516**	**0.8437**
無	0.8530	0.8462

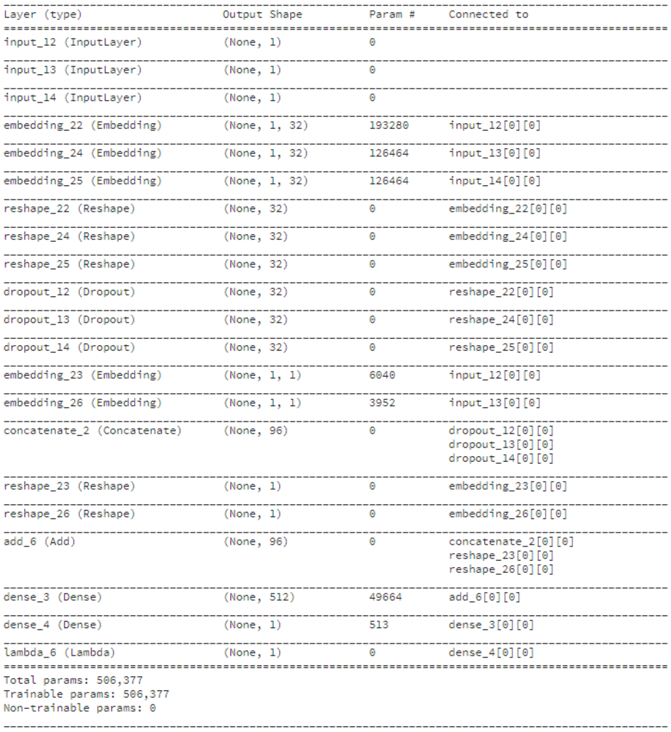
在考慮 bias 的情況下，是會對 loss 有些微的改善，可推得訓練模型時仍有一些未考慮到的 feature，或是資料存有 noise，透過增加 bias 項可以改善這些未考慮/未排除的情況。

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



利用表現最好的模型（loss = 0.8516/0.8437）中的 embedding，降維後取直覺認為應為類似分類的為同一群，並設定另一類別為對照組，如「Action 跟 Adventure」對比「Horror 跟 Crime」。

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。
   答：
   原先僅考慮 User ID 跟 Movie ID 進行 embedding，讓模型自己透過 rating 去解決各使用者及電影間的特徵關係。於本題欲嘗試增加 "Genres"特徵，並同樣透過 embedding 轉換成向量，藉由 keras.layers.concatenate 結合，後續則改接上 DNN 的結構（如圖一），訓練方式維持不變，再與原先利用 MF 方式實作的精度作比較，結果如下表。



```
____________________________________________________________________________________________________
Layer (type)                     Output Shape         Param #     Connected to
====================================================================================================
input_12 (InputLayer)            (None, 1)            0
____________________________________________________________________________________________________
input_13 (InputLayer)            (None, 1)            0
____________________________________________________________________________________________________
input_14 (InputLayer)            (None, 1)            0
____________________________________________________________________________________________________
embedding_22 (Embedding)         (None, 1, 32)        193280      input_12[0][0]
____________________________________________________________________________________________________
embedding_24 (Embedding)         (None, 1, 32)        126464      input_13[0][0]
____________________________________________________________________________________________________
embedding_25 (Embedding)         (None, 1, 32)        126464      input_14[0][0]
____________________________________________________________________________________________________
reshape_22 (Reshape)             (None, 32)           0           embedding_22[0][0]
____________________________________________________________________________________________________
reshape_24 (Reshape)             (None, 32)           0           embedding_24[0][0]
____________________________________________________________________________________________________
reshape_25 (Reshape)             (None, 32)           0           embedding_25[0][0]
____________________________________________________________________________________________________
dropout_12 (Dropout)             (None, 32)           0           reshape_22[0][0]
____________________________________________________________________________________________________
dropout_13 (Dropout)             (None, 32)           0           reshape_24[0][0]
____________________________________________________________________________________________________
dropout_14 (Dropout)             (None, 32)           0           reshape_25[0][0]
____________________________________________________________________________________________________
embedding_23 (Embedding)         (None, 1, 1)         6040        input_12[0][0]
____________________________________________________________________________________________________
embedding_26 (Embedding)         (None, 1, 1)         3952        input_13[0][0]
____________________________________________________________________________________________________
concatenate_2 (Concatenate)      (None, 96)           0           dropout_12[0][0]
                                                                  dropout_13[0][0]
                                                                  dropout_14[0][0]
____________________________________________________________________________________________________
reshape_23 (Reshape)             (None, 1)            0           embedding_23[0][0]
____________________________________________________________________________________________________
reshape_26 (Reshape)             (None, 1)            0           embedding_26[0][0]
____________________________________________________________________________________________________
add_6 (Add)                      (None, 96)           0           concatenate_2[0][0]
                                                                  reshape_23[0][0]
                                                                  reshape_26[0][0]
____________________________________________________________________________________________________
dense_3 (Dense)                  (None, 512)          49664       add_6[0][0]
____________________________________________________________________________________________________
dense_4 (Dense)                  (None, 1)            513         dense_3[0][0]
____________________________________________________________________________________________________
lambda_6 (Lambda)                (None, 1)            0           dense_4[0][0]
====================================================================================================
Total params: 506,377
Trainable params: 506,377
Non-trainable params: 0
____________________________________________________________________________________________________
```

圖一

	Public	Private
原先 MF	0.8516	0.8437
DNN (考慮 Genres)	0.9124	0.9069