COMP4332 Project 3 – Rating Prediction (Group 14)

Members: CHEN, Jiawei (20763842), Leung King Suen (20770625), Kwong Ka Lok (20772439)

Introduction

We utilized the code from tutorial 8 as the base of our project, the model was implemented as a Neural Collaborative Filtering (NCF) Model with two instantiation methods: **GMF Layer** and **MLP Layer**. The **GMF Layer** instantiation computes the recommendation score (ratings) between a pair of user and item using dot product of their embeddings, which is equivalent to matrix factorization model for recommendation. The **MLP Layer** instantiation concatenates the user's and item's embeddings, then feeds the concatenated vector into a MLP to calculate the recommendation score. Adoption of MLP equips the model with high flexibility and non-linearity to effectively learn the interaction between user and item latent features. We first use the MLP instantiation and yield a **0.86** RMSE score on the validation set and we used this as a baseline for further improvement.

Approach One: GMF Layer (Dot Product)

The baseline model utilized the MLP Layer to calculate the recommendation score, so the first thing we tried was to find out whether using the GMF Layer (Dot Product) would have a better performance than using MLP Layer. After several trials with different embedding size settings, we found that this architecture is prone to overfitting, and we cannot get it converge for our validation set. The RMSE score on validation set was extremely high (e.g. 2.625).

Approach Two: Parameter Tuning on MLP Layer

Since the previous approach did not work out, we decided to focus on the MLP

Layer and tuning its parameters. The following table shows the result:

Embedding Size	**Dropout Rate**	**RMSE Score (on Validation Set)**
16	0	**0.8354 (final model)**
	0.15	0.8432
	0.3	0.8544
50	0	0.8408
	0.15	0.8455
	0.3	0.8462
64	0	0.8419
	0.15	0.8461
	0.3	0.8538
128	0	0.8457
	0.15	0.8472
	0.3	0.8560

As the figure shows, we experimented with two parameters: embedding size and dropout rate between the forward layers. We found that as the embedding size increases, the model's performance degrades slightly and increasing the dropout rate would also degrade the performance. We yield a best performance of 0.8354 RMSE score on validation set with Embedding Size=16 and Dropout Rate=0. We successfully improved the performance by 0.0246 compared to the baseline. This setting and model architecture was used as our final model to predict the scores on test set.

Approach Three: NeuMF Layer

In addition to using the GMF and MLP Layer separately, we also tried to add a NeuMF Layer which concatenates the output embeddings from these two layers and use the concatenated embeddings as final output for calculating the loss. We hope this could leverage the power from both layers and improve the model's performance. After training, we found this approach yield a similar performance compared to our second approach, the combination of both GMF and MLP Layer did not help much. Thus, we decided to use the second approach as our final model architecture.