Movie Recommender System

Mingyue Jian
Computer Science Department
Durham University
Durham, United Kingdom
knpv42@durham.ac.uk

***Abstract*—This paper is a coursework for Durham University Level 3 Recommender Systems. The project will implement a non-personalised recommender system and apply neural collaborative filtering for a personalised recommender system.**

***Index Terms*—recommender systems, deep learning, collaborative filtering**

I. INTRODUCTION

A. Domain of application

The film industry has experienced substantial growth in recent years, resulting in a proliferation of noteworthy productions. The project focuses on the domain of movies. A well-designed movie recommender system has the potential to significantly impact the film industry by improving the ease of discovery of quality films, providing personalised recommendations to users, enhancing user engagement, and mitigating the information overload issue.

B. Purpose

This project aims at providing two different types of movie recommender system - personalised and non-personalised, and comparing the differences between them. To achieve better performance, the personalised system is a collaborative filtering system that applies a deep learning technique (NCF) [1] with generalised matrix factorisation (GMF) and multi-layer perceptron (MLP) model architecture components. The non-personalised system is a system to output the highest ranked movies based on the database, conditioning on the average rating, watch time and diversity of the genres. The final presentation of the two types of systems are in the "Meowie Recommender System" Application.

II. METHODS

A. Data description

The dataset selected for the project is the 10M10K dataset from MovieLens [2]. The dataset comprises 10,000,054 ratings for 10,681 movies, contributed by 71,567 unique users, with a sparsity level of 98.7%. All users selected for the dataset had rated at least 20 movies. The features of the dataset consists of 'UserID', 'MovieID', 'Rating', 'Timestamp', 'Tag', 'Title' and 'Genres'. The choice of this dataset is justified by its benchmark status in the field of recommender systems research, its well-established reputation, and its convenient accessibility.

B. Data preparation

The first part of data preparation will focus on invalid data. After data exploration, data cleansing will focus on the removal of users who gave extreme values as well as the removal of users or movies that have a small amount of interactions, to be more precise:

- Remove users who gave extreme ratings
- Remove users who rated less than 50 movies
- Remove movies which were rated by less than 100 users

Cleansing this data will enhance the performance of the system.

The second part will focus on rescaling the ratings. The original ratings are presented as floats, starting from 0.5, with 0.5 increments up to 5.0. The rescaled data is the double of the original data, and the data type changes from floats to integer. For a large-scale dataset processing, the rescaling is significant as it'll reduce the memory cost of the machine while not changing the data representation. After training, the rating and prediction will be scaled back down to the original range, to ensure the accessibility of the data for users.

C. Data presentation and feature selection

After data cleaning, the sparcity of the dataset is 96.46%. Table I presents the dataset for training after data cleaning.

	User	Movie	Rating	Genres
Count	43,435	5,855	8,991,513	18

TABLE I: Data presentation for the recommender system.

The selected features for the prediction includes 'UserID', 'MovieID', 'Title' and 'Rating'. Ratings are explicit feedback from the users, it'll be used as output prediction as well. The features for evaluation comprised of the ones for prediction as well as 'Genres'. The training data for the personalised recommender system would be the data where users have rated a given movie in a User-Item matrix, i.e. a three-tuple of (u, i, r_{ui}), where u, i are the user and item IDs and r_{ui} is the rating, which is the target. The original paper changes the rating to implicit feedback (binary representation of whether the user has had contact with the movie) as prediction target. This project selects explicit feedback for better accuracy of the prediction. The reason lies in that a user having had contact with an item does not mean the user has interest in the item, so it might produce lower accuracy.

The training data for the non-personalised recommender system would be the data used for the personalised system combined with the number of ratings and the number of genres of each movie.

The dataset will be divided into training and testing subsets for evaluation purposes, in a 7:3 ratio. The division will be performed using a consistent random seed to ensure the validity of the evaluation results. For the non-personalised system, the split will be between users, such that a user's entire set of ratings are in either the training set or the testing set. For the personalised system, the split is between ratings, so a given user will likely be represented in both sets.

D. Personalised Recommendation techniques

1) Personalised techniques selection: There are two types of techniques for personalised filtering: content-based and collaborative filtering. Content-based filtering uses the attributes of items to make recommendations, such as genres to movies. It generates recommendations based on the similarity between an item and other items in the database. That is why content-based filtering provides less diversity.

Collaborative filtering, on the other hand, is a technique that uses the past behavior and preferences of users to make recommendations. It generates recommendations based on the similarity between a user and other users in the database, and their past interactions with items. It is better suited for use with a large-scale dataset, as it is based on the user profiles, rather than the characteristics of items.

For a large-scale dataset like MovieLens 10M10K, this project selects collaborative filtering as it provides greater scalability, better handling of the cold start problem, more personalised recommendations, and a more diverse set of recommendations.

Collaborative Filtering is usually divided into two types in terms of techniques. The project will apply a model-based collaborative filtering technique. Due to the extreme sparsity of the data, using memory-based techniques would greatly reduce the prediction performance.

Compared to the most popular collaborative filtering recommendation approach - matrix factorisation (MF) - a deep neural network based approach provides better predicting performance. MF decomposes a feedback matrix Y into a user factors matrix P and item factors matrix Q, which is too easy to be under-fitting and lead to the cold-start problem. In other words, if an item is not seen during training with MF technique, the system can't create an embedding for it and can't query the model with this item. To address such shortcomings, we apply deep learning approach with the Neural Collaborative Filtering (NCF) model. Instead of using just the scalar product to combine P and Q, NCF also uses a deep neural network to combine P and Q, which can have more feature crossover and non-linear operations.

2) Personalised techniques description: The training data for the model will be the item in the User-Item matrix R, where users have rated a given movie. The ratings for the item will be used to tune the loss function. After training, the NCF model takes each user ID and each item ID as input, both of them will be presented as one-hot vectors as the type of data is categorical, and output a prediction rating for an unseen movie for the input user. The prediction for the system will be conducted by feeding the item ID of each element in a list of unseen movies for the target user, each element will be in the form of a one-hot vector. All prediction rating and relevant data (e.g. movieID) will be stored in a recommendation list. Recommendation for the users will be selected by the top 30 movies in the recommendation list.

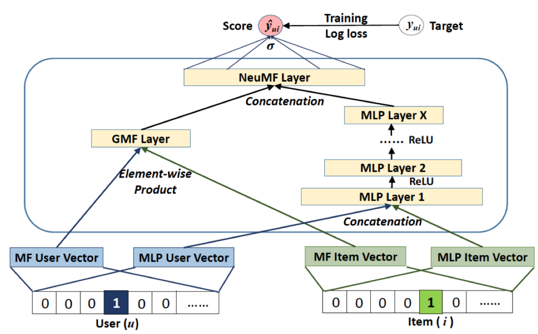
Inside the model we will apply MLP and GMF model instances. We define the NCF's model based on the paper [1] as:

$$\hat{r}_{ui} = f(\mathbf{P}^T \mathbf{v}_u^U, \mathbf{Q}^T \mathbf{v}_i^I | \mathbf{P}, \mathbf{Q}, \Theta_f), \tag{1}$$

where $\mathbf{P} \in \mathbb{R}^{M \times K}$ and $\mathbf{Q} \in \mathbb{R}^{N \times K}$, denote the latent factor matrix for users and items, respectively; and Θ_f denotes the model parameters of the interaction function f.

Graph 1 is a visualisation of the process.

Graph 1: Neural matrix factorisation model visualisation from [1]



Where each MLP Layer is a linear transformation separated by the ReLU activation function, and the GMF Layer is a simple elementwise product. The NeuMF Layer is a final linear transformation, followed by a sigmoid function to reach the output layer.

We will apply Mean Squared Error (MSELoss) for evaluation. We defined the loss function as follows:

$$MSELoss = \sum_{u,i \in \text{data}} (t_{ui} - \hat{r}_{ui})^2$$

Where u, i are user and item IDs and t_{ui} is the value of r_{ui} scaled to the range $[0, 1]$ to match with the output of the sigmoid function.

E. Non-Personalised Recommendation techniques

A non-personalised recommender system does not take into account personal information or preferences of an individual user. It generates recommendations for a group of users based on different features, without considering the interests of each user. The features selected for the non-personalised system for the project are the average rating of the movies, the number of ratings for each movie as well as the diversity of genres of the movies. The higher each of these feature is, the higher the ranking of one movie is. The final ranking will be produced based on the average rating of the movies, then, when having the same average rating, the ranking will then be decided by the number of ratings of the movie, and similarly, the next decider is the the diversity of genres of the movies.

F. Evaluation methods

Two types of methods are applied for evaluation. They are accuracy of rating predictions and diversity. For accuracy of rating predictions, MSE, RMSE, Precision and Recall metrics are applied.

MSE is used to assess the difference between predicted rating and ground truth rating. RMSE is also used, because it penalises larger errors more - similar ratings are interchangable, but large differences could be problematic for recommendation. This is the most important metric for accuracy.

$$MSE = \sum_{i=1}^{D}(x_i - y_i)^2$$

$$RMSE = \sqrt{\sum_{i=1}^{D}(x_i - y_i)^2}$$

Precision checks the proportion of output recommendations which are actually good, and recall checks the proportion of good movies which are recommended. These are the important metrics for assessing the quality of recommendations, whereas those above only focus on ratings. Good movies are considered to be those with a rating of at least 3.5.

$$Precision = \frac{|\text{Good recommendations}|}{|\text{All recommendations}|}$$

$$Recall = \frac{|\text{Good recommendations}|}{|\text{All good movies}|}$$

Mean cosine similarity calculates the similarity in the genre vectors between (pairs of) recommended movies, which measures the diversity of recommendations.

$$MeanCosine = 1 - \frac{1}{n}\sum_{i=1}^{N}\sum_{j=i+1}^{N}\frac{\boldsymbol{x}_i \cdot \boldsymbol{x}_j}{\|\boldsymbol{x}_i\| \cdot \|\boldsymbol{x}_j\|}$$

III. Implementation

Both recommender systems were implemented in Python3. The NeuralMF model was trained for 100 epochs using Adam's optimiser with a learning rate of 0.001. Both models were configured to provide 10 recommendations.

A. Input interface

The recommendation system is accesible by a Python script 'user.py'. After the data loading, users can select whether they want to know more about the recommender system, which recommender system techniques are applied, how the models are trained, and how the data would be used. After that, users would be directed to the selection of recommender systems. If the selection is personalised recommender system, the user is required to input their user ID to identify themselves to the system; if the selection is the non-personalised system, the user does not need to input anything.

B. Output interface

After receiving input, the system prints the titles of the top 10 recommended movies to standard output in a numbered list.

IV. Evaluation

Both systems are evaluated based on their respective test set (as per Section II-C), which is split after data cleaning. The output prediction from the personalised system will be scaled back to the 1 - 10 range. Evaluations are based on the metrics outlined in Section II-F.

A. Evaluation metrics' results

Table II displays the results of evaluation for both systems. The better result for each metric is highlighted with bold.

system	MSE	RMSE	Precision	Recall	Diversity
Personalised	$\mathbf{2.4 \cdot 10^{-4}}$	**0.02**	**1**	$2.4 \cdot 10^{-4}$	0.59
Non-Personalised	2.36	1.53	0.99	**0.01**	**0.97**

TABLE II: Evaluation results for the both systems.

B. Comparison

According to Table II, the personalised system performs better in MSE, RMSE and Prcision, but worse in Recall and Diversity. This indicates that the personalised system is better in accuracy, which is correctly predicting how highly a user will rate a given movie and so giving more high-quality recommendations, whereas the non-personalised system is better at producing more diverse recommendations.

V. Conclusion

In conclusion, this project implemented two recommender systems - a personalised recommender system and a non-personalised one, using MovieLens 10M10K dataset. A Neural collaborative filtering technique is applied for the personalised system, and the non-personalised system gives recommendations based on the average ratings, the number of ratings of a movie and the number of genres the movie includes. Based on the evaluation results, we draw the conclusion that while the non-personalised system produces vastly more diverse recommendations than the personalised one, the personalised system does produce recommendations which are more accurate to the specific user's tastes. Additionally, the personalised system takes considerably more time and compute power to produce recommendations than the non-personalised system does, and would require architectural changes to allow it to be re-trained for new user registrations and new movie releases.

References

[1] He, X., Liao, L., Zhang, H., Nie, L., Hu, X., and Chua, T. S. “Neural collaborative filtering,” In Proceedings of the 26th international conference on world wide web, pp. 173–182, April 2017.

[2] F. Maxwell Harper and Joseph A. Konstan “The MovieLens Datasets: History and Context,” ACM Transactions on Interactive Intelligent Systems (TiiS) 5, Article 19 , 19 pages, December 2015.