Weighted Hybrid Movie Recommender System using Collbarative and Content-Based Filtering

I. Introduction

The rise in popularity of online streaming platforms such as Netflix, Disney+ etc. mean that there is a great demand in constructing recommender systems that provide accurate recommendations to users, whilst also engaging them and encouraging them to explore new content.

The recommender system (RS) stated in this project is in the movie domain and will be used to provide users personalised recommendations for movies based on a combination of: movies with a similar content to what the user positively interacted with using a TFIDF technique and interactions from similar users using an SVD technique.

This paper aims to minimise RMSE of the recommendations, whilst also ensuring relevant but also diverse recommendations are also included to encourage movie exploration.

Finally, the personalised RS will be compared with a non-personalised popularity based RS using appropriate evaluation metrics.

II. Methods

A. Data

The MovieLens 10M dataset (https://grouplens.org/datasets/movielens/10m/) contains 10,000,054 ratings and 95,580 tags for 10,681 movies, given by 71567 users of the MovieLens online movie recommender service [1]. The dataset is split into three files: movies.dat, ratings.dat and tags.dat. Altogether, they provide the following features: user ID; movie ID; title; rating which is given on a 5-star scale with increments of 0.5; tags which are user-generated metadata in the form of a phrase or singular word; genre which is presented in a pipe-separated list; and timestamp.

Timestamps were removed from the data as the RS to be implemented is only interested in movie content and user ratings, and rows with missing values were dropped.

Although other experiments removed movies with a low number of ratings, e.g. 3000 ratings [2], to filter out noise from obscure movies, this project did not filter on ratings in order to utilise the available data and maximise the diversity and possible novelty of recommendations to aid user exploration.

The tags column contained metadata from users which give useful insight into the movie contents and properties. Punctuation, numbers and symbols were filtered out from the tags. Genres were converted from pipe-separated values to uppercase values separated by spaces to have a similar structure to tags. Both these columns were kept to help suggest movies with similar content to users.

The data was combined and split into 80% training data (suggested by many including [3]) and 10% was used for validation and 10% for testing.

B. Recommendation Techniques

Collaborative filtering (CF) uses interactions from similar users to provide predictions [4] whereas content-based filtering (CBF) utilises item content [4]. Although studies have shown that CF approaches tend to yield more accurate results than CBF [5], content-based filtering has the advantage of being more robust to the cold-start problem and popularity bias [6] because it does not have to wait for data from other users. However, the content-based recommendations can lack diversity as similar items are found based on their content, unlike collaborative filtering where items are suggested based on collective user ratings and so has a greater chance of being more diverse. Due to the advantages both techniques provide, this project will use a hybrid approach that combines the two.

C. Popularity

To assess the personalised RS performance, a popularity RS will be used as the baseline since its a popular non-personalised RS used in online streaming platforms. It outputs recommendations based on the most popular movies users interacted with. The ratings for movies are calculated using IMDB's weighted rating technique [7]

$$WeightedRating = (\frac{v}{v+m} \cdot R) + (\frac{m}{v+m} \cdot C) \quad (1)$$

where v is the number of movie votes, m is the minimum number of votes required to be added to the list, R is the average rating of the movie and C is the mean vote across the list [7].

D. SVD

For the CF RS, a Singular Value Decomposition (SVD) technique was implemented. Although memory-based CF models are more explainable, model-based techniques like SVD are better scalable to large volumes of data [8] like the MovieLens dataset.

$$\hat{r}_{u,i} = \mu + b_u + b_i + q_i^T p_u \quad (2)$$

The SVD technique creates a user-movie matrix and splits this into separate user and movie embeddings with m latent factors and the dot product is calculated. The technique used for this project also takes into account the overall average rating (μ), and the user and item bias denoted as (b_u) and (b_i) respectively to calculate ratings [eq. (2)].

To find the optimal value for the number of latent factors, the SVD model was trained on the training set with 20, 40, 60, 80 and 100 latent factors and stochastic gradient descent was performed for 20 epochs each to minimise the following regularised square error:

$$\sum_{r_{ui} \epsilon R_{train}} (r_{ui} - \hat{r}_{ui})^2 + \lambda(b_i^2 + b_u^2 + \|q_i\|^2 + \|p_u\|^2) \quad (3)$$

The resulting models were evaluated on the validation set using RMSE and the model with 80 latent factors was selected due to its low error score.

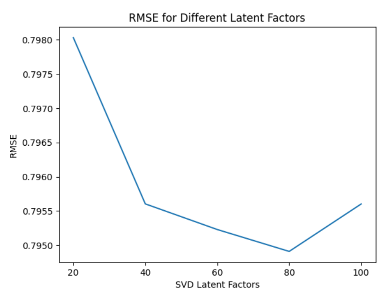

Fig. 1. Graph showing how RMSE for SVD on the validation set changes with different latent factors

Rating predictions can then be generated based on the calculated rating value in the new user-movie matrix.

E. TF-IDF

For the CBF RS, a TF-IDF approach was taken to utilise the corpus of tags available in the dataset that could give meaningful information about movie content [6]. The data exploration phase revealed that genres did not appear in the majority of movie tags. Since the genre can be indicative of a movie's content, they were concatenated with tags and a new 'document' column [6] was created that contained a corpus of genre and tag keywords for each movie. Duplicate movie rows were dropped and TF-IDF was applied on the 'document' column to generate a matrix with movie profiles.

To generate recommendations from the CBF approach, a user profile for a given user was created by subtracting movie profiles where user negatively interacted with (rating of less than or equal to 3) and adding movie profiles that the user positively interacted with (rating of greater than 3). The user profile was then compared to the interacted movie profiles in the TF-IDF matrix to find the top N movie predictions. The predicted rating was simply calculated by multiplying the similarity score by 5, which assumes that the user rating is directly proportional to the content similarity of the movie and other movies they have positively interacted with.

F. Hybrid

Since both the CBF and CF RSs have desired properties that help achieve the goal of this project, a weighted scheme approach was used where both RSs were used to generate candidates and a linear weighting scheme was used to score them. Then the top 5 (see [Sec. IV]) recommendations from the sorted weighted combination were used to recommend unrated movies for the user. A weighted scheme enables greater control over the contribution of each model, allowing optimisation over chosen metrics.

G. Evaluation Metrics

RMSE and diversity were used to evaluate the RS.

RMSE helps evaluate the accuracy of the RS and was chosen over metrics such as MAE because it penalises extreme values [9].

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(\hat{r_i} - r_i)^2}{N}} \quad (4)$$

Large errors are particularly undesirable in this domain because the user is only shown a small number of recommendations at a given time and a movie with a falsely high rating can appear in the list of recommendations, which means the user has a high chance of interacting with that items and losing trust in the system.

The diversity of n movie recommendations is measured as the average dissimilarity between all pairs in the recommendation set [10] and is given as follows:

$$Diversity(c_1, ...c_n) = \frac{\sum_{i=1..n}\sum_{j=i..n}(1 - Similarity(c_i, c_j))}{\frac{n}{2} * (n-1)} \quad (5)$$

Since movie profiles are already generated in the CBF RS, the similarity measure between c_i and c_j was calculated by retrieving the movie profiles from the TFIDF matrix and calculating the cosine similarity between the two profiles.

This metric was chosen because it indicates the level of exploration of movies the system provides to users which is desirable in the movie domain. For example, a low diversity score suggests that the system overfits to user preference and may filter out items that are different to the users' taste, whereas a high diversity score implies that the movies recommended are highly varied in terms of similarity and are random.

III. Interface

Upon signing in with a unique user ID and password, the user is presented with a list of top 5 recommendations in a tabular format as this is the main purpose of the system. Although CBF helps prevent the cold start problem resulting with a low number of users, its still prone to not being able to recommend movies accurately if the number of movies the user rated is low or even non-existent. To combat this issue, the RS uses the popularity model to recommend non-personalised movies for users who have rated less than 10 movies and switches to the hybdrid model for users who have rated more than this. Both users, i.e. those who receive personalised and non-personalised recommendations, are also given an explanation of results for explainability purposes and have the ability to: generate the next 5 predictions, find out more details about how their data is being used for ethical reasons, and also have the option to logout or exit the system as shown in Fig. 2. Users who receive personalised recommendations also have the option to view the top 5 popular recommendations as well using the popularity RS.

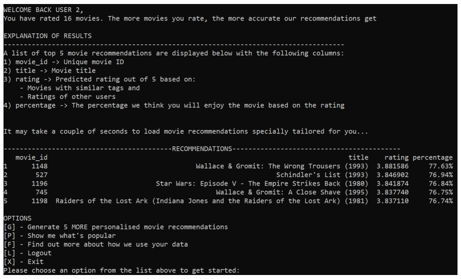

Fig. 2. Image showing user interface for personalised users

IV. Evaluation

Table 1 comparison of RMSE and diversity scores for Popularity and Hybrid RS

	RMSE	Diversity@10	Diversity@5
Popularity	0.9408	0.9349	0.9636
Hybrid	0.8090	0.9351	0.9003

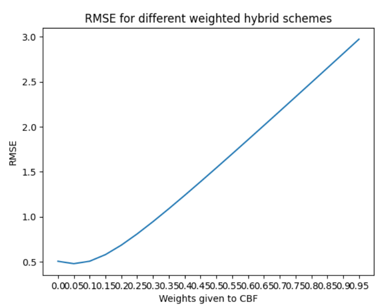

Fig. 3. Graph showing how RMSE on the validation set changes with hybrid scheme weights

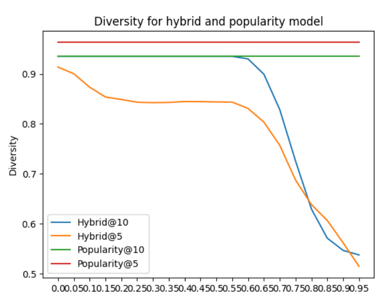

Fig. 4. Graph showing how diversity changes with hybrid scheme weights compared to the diversity of the popularity model

To find the best weighting scheme, the hybrid model was evaluated on both metrics for weights ranging from 0.00 to 0.95 in increments of 0.05 as seen in [Fig. 3 and Fig. 4].

The diversity metric was applied on the top 5 and top 10 recommendations for a sample of a 1000 users (limited due to computational limitations) and the average was calculated. The diversity score for both the popularity model and hybrid model at 5 and 10 recommendations were both very high. The diversity for the hybrid model remains high until the weighting for CBF is around 0.55, after which the diversity drops significantly [Fig. 4]. This is expected, because the CBF provides recommendations based on movie similarity and these recommendations get a higher rating as the weighting for CBF increases.

Although a balanced diversity score would be desirable, the RMSE for the hybrid model on the validation set is optimal when the weighting for CBF is 0.05 and increases rapidly for higher weights (e.g. the RMSE is more than 3.5 times higher for a weighting of 0.55 than 0.05). Since the RMSE is a more desired metric for this domain, and since the diversity is most balanced when the top 5 recommendations are shown, the weighting chosen was set to 0.05 for CBF and 0.95 for CF, and only the top 5 recommendations were chosen to be outputted by the RS interface.

However on the test set, the hybrid model had a higher RMSE of 0.8090 [Tab. 1] compared to its score of 0.4780 [Fig. 3] on the validation set at 0.05 weighting for CBF. This suggests that the model significantly overfit to the data, possibly as a result of a high number of latent factors used.

It is also important to note that the MovieLens data was very sparse which can negatively impact the rating predictions, particularly for the CF model.

Overall, although the personalised RS overfit to the training data, the study has shown that the hybrid model can still predict outperform the baseline popularity model in terms of accuracy dictated by RMSE and provide a more balanced set of diverse recommendations.

V. Conclusion

In this project, a hybrid RS that used a weighted scheme for combining a CBF model using a TFIDF approach and a CF model using a SVD approach was compared to a non-personalised popularity RS using RMSE and diversity as metrics. The results showed that the personalised hybrid approach had better accuracy and a more balanced diversity metric, thus providing accurate recommendations whilst encouraging exploration as required in the movie domain.

A. Limitations

A key limitation was the sparsity of data that could have led to incorrect rating predictions, particularly for the CF model. Also computational limitation was also an issue that prevented the use of more advances techniques such as SVD++ [11] that takes into account implict feedback.

B. Future Work

Future work could address issues to reducing the overfitting of the model, and explore the hybrid weighting scheme at smaller intervals around 0.05 for CBF where the RMSE was lowest.

REFERENCES

[1] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. ACM Transactions on Interactive Intelligent Systems (TiiS) 5, 4, Article 19 (December 2015), 19 pages. DOI=http://dx.doi.org/10.1145/2827872

[2] J. Ivarsson, M. Lindgren, 'Movie recommendations using matrix factorization'. 2016.

[3] U. Kuzelewska, 'Clustering algorithms in hybrid recommender system on movielens data', Studies in logic, grammar and rhetoric, vol. 37, no. 1, pp. 125–139, 2014

[4] P. B. Thorat, R. M. Goudar, and S. Barve, 'Survey on collaborative filtering, content-based filtering and hybrid recommendation system', International Journal of Computer Applications, vol. 110, no. 4, pp. 31–36, 2015.

[5] J. F. G. da Silva, N. N. de Moura Junior, and L. P. Caloba, 'Effects of Data Sparsity on Recommender Systems based on Collaborative Filtering', 2018 International Joint Conference on Neural Networks (IJCNN). IEEE, Jul. 2018. doi: 10.1109/ijcnn.2018.8489095.

[6] A. Lineberry and C. Longo, "Creating a hybrid content-collaborative movie recommender using deep ...," Creating a Hybrid Content-Collaborative Movie Recommender Using Deep Learning. [Online]. Available: https://towardsdatascience.com/creating-a-hybrid-content-collaborative-movie-recommender-using-deep-learning-cc8b431618af. [Accessed: 05-Nov-2022].

[7] "Movie Recommender Systems," kaggle.com. https://www.kaggle.com/code/rounakbanik/movie-recommender-systems/notebook (accessed Dec. 22, 2022).

[8] J. Bobadilla, R. Bojorque, A. Hernando Esteban and R. Hurtado, "Recommender Systems Clustering Using Bayesian Non Negative Matrix Factorization," in IEEE Access, vol. 6, pp. 3549-3564, 2018, doi: 10.1109/ACCESS.2017.2788138.

[9] T. Rackaitis, "Evaluating Recommender Systems: Root Means Squared Error or Mean Absolute Error?," Medium, May 31, 2019. https://towardsdatascience.com/evaluating-recommender-systems-root-means-squared-error-or-mean-absolute-error-1744abc2beac

[10] K. Bradley, 'Improving Recommendation Diversity', Proc. AICS '01, 09 2001.

[11] S. Jiang, J. Li and W. Zhou, "An Application of SVD++ Method in Collaborative Filtering," 2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), 2020, pp. 192-197, doi: 10.1109/ICCWAMTIP51612.2020.9317347.