International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-11 Issue-8, July 2022

Development of Online Movie Recommendation System based on Neighborhood-based Collaborative Filtering

Phan Thi Ha, Trinh Thi Van Anh

Abstract: ***The recommendation system integrated in movie streaming provides relevant information to viewers predicted by viewers' past behaviors. There are basically two methods, Content-Based Filtering and Collaborative Filtering. In this article, our focus is on the second method which is based on memory, namely Neighborhood-based Collaborative Filtering (NBCF), to make movie recommendations to users given users' information. Simultaneously, we have built an online movie website and integrated the movie recommendation system based on NBCF to assist users in movie selection. In the process of building the website, apart from building diagram of movie recommendation system's functions, class diagram of movie recommendation function, sequence diagram of movie recommendation function, we also build a user-recommended movie model based on the Movies Lens[9] dataset for a fairly high accuracy, which is 99%.***

Keyword: Neighborhood-based Collaborative Filtering (NBCF), recommendation system, Item_KNN.

I. INTRODUCTION

Nowadays, online movie streaming platforms have allowed viewers to access many movies at the same time. Hence, viewers find it difficult to pick the best movie to watch because there are simply too many movies on display. Therefore, an integrated movie recommendation system on online movie websites is extremely important, making it easier for viewers to select their favorite movie from a given list of recommended movies.

Recommendation systems often use many different algorithms, which can basically be divided into two large groups [1,2], Content-Based Filtering and Collaborative Filtering as well as hybrid approaches that combine collaborative and content-based methods.

Content-Based Recommendation System uses attributes of products, such as name, manufacturer, price, description... to recommend similar products. The feature of this method is that the construction of a model for each user does not depend on other users, but often depends on the characteristics of the product. In which, the collaborative filtering recommendation system uses the similarity between users or between products in the system to the user's interest in a product, or to make recommendations of a new product to users. It then matches users with interests and preferences respectively by detecting the similarity between user profiles to generate recommendations. There are two types of collaborative filtering techniques: memory-based and model-based. There are also other types such as context- , properties-, behavior-, knowledge-based [3][4]. Currently, there have been many studies on the use of collaborative filtering recommendation system to analyze and synthesize data in education, predict learning outcomes and suggest learning strategies for students, give musical advices…[5,6,7,8].

In this paper, we will use the memory-based Neighborhood-based Collaborative Filtering recommendation algorithm in a movie streaming website for users to easily find the movies they want. This algorithm will be presented in Part 2. In Part 3, we will explain the incorporation of NBCF into developing a movie recommendation system.

II. NEIGHBORHOOD-BASED COLLABORATIVE FILTERING

Collaborative recommendation systems (or collaborative filtering systems) aim to predict utility of products for a specific user based on products previously rated by other users. The main idea of collaborative recommendation technique is to use information about the previous behavior of existing users in the system to predict which products current users will likely enjoy the most and hence will use.

For the collaborative filtering recommendation system, the input is a matrix representing the user's rating for the products, element of which is (uk, ih) with values ranging from 0 to 5 (the number of stars that uk users rate product ih and the output will generate a prediction as a number indicating the current user's liking or disliking a certain product, or a list of n suggested products). The list generated does not contain the items the current user has used. Neighborhood-based collaborative recommendation systems work by counting the items in common that two users have viewed for each pair of users in the system, or number of users that have viewed the same pair of items. The similarity between two people or items is then calculated. Two people who viewed a large number of the same items have similar interests.

Manuscript received on 08 June 2022 | Revised Manuscript received on 15 June 2022 | Manuscript Accepted on 15 July 2022 | Manuscript published on 30 July 2022.

***** Correspondence Author

Dr. Phan Thi Ha*, Lecturer, Faculty of Information Technology, Posts and Telecommunications Institute of Technology (PTIT), Ha Noi, Vietnam, and Department of Computing Fundamental, FPT University, Hanoi, Vietnam. E-mail: hapt@ptit.edu.vn

Trinh Thi Van Anh, Lecturer, Faculty of Information Technology, Posts and Telecommunications Institute of Technology (PTIT), Ha Noi, Vietnam, and Department of Computing Fundamental, FPT University, Hanoi, Vietnam. E-mail: vanh22@yahoo.com

We need to find the pairs of users with the most similar interests, or the pairs of items viewed by most users. Those pairs of users/items are called “the nearest neighbors”. The two main approaches of neighborhood-based recommendation systems are user- and item-based nearest neighbor recommendations.

In collaborative filtering, there are usually two approaches according to the K neighboring model: the user-based system (User_KNN) - a prediction based on the similarity between the users; item-based system (Item_KNN) - a prediction based on similarity between products. To calculate similarity between users, there are two commonly used similarity calculation methods: Pearsn and Cosine correlation. Similarity values range from -1 and 1. Usually not all users in the data are considered when calculating user similarity, but only the k most similar users are considered. The similarity calculation formula to calculate the similarity between two users ui and uj is consine similarity, formula [1]. After calculating the similarity between users, we can predict the rating of user u on product i. Here we will predict according to the nearest K- User , the more K, the more accurate the prediction will be, the formula [2].

$$cosine\_similarity(u_1, u_2) = \cos(u_1, u_2) = \frac{u_1^T u_2}{\|u_1\|_2 . \|u_2\|_2} \quad [1]$$

$$\hat{y}_{i,u} = \frac{\Sigma_{u_j \in N(u,i)} \bar{y}_{i,u_j} sim(u,u_j)}{\Sigma_{u_j \in N(u,i)} |sim(u,u_j)|} \quad [2]$$

- u: user that the system needs to predict the number of stars that user might rate for the movie i.
- N(u, i): the set of k users with the greatest similarity to user, in which k users have made a rating for movie i in the past.
- sim(u, u_j): the user's similarity value compared to the j^{th} user.
- y_i, u_j : j^{th} user's standardized rating for movie i.

III. INCORPORATING COLLABORATIVE FILTERING INTO DEVELOPING MOVIE RECOMMENDATION SYSTEM

A. Diagram of movie recommendation system’s functions

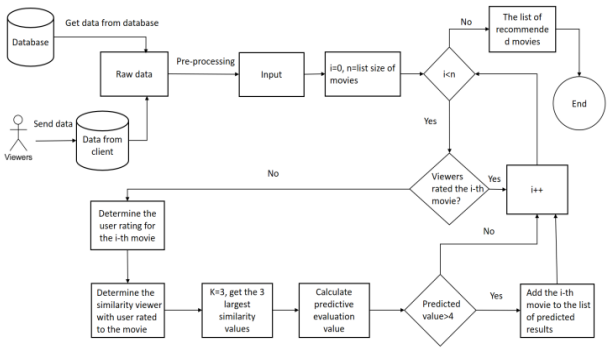

Figure [1]. Diagram of movie recommendation system’s functions

• Raw input will be derived from database including movie list, user list, rating list. In addition, the raw input will be taken from the user request sent from the client including the account name and selected movie.

• After the raw data is obtained, it will be formatted and converted to Map<account name, movie list and account's rating>

• The next steps will be to predict the number of stars rated by viewers for the first movie:

- Check through each movie in the system
- Check if the movie has been rated by viewers

✓ If already rated, increment i returns to the beginning of the loop

✓ If not yet rated, predict the viewer's rating for the movie by following these steps:

✓ Identify the users who have rated the selected movie

✓ Determine k values of maximum similarity of viewers compared to other users according to the formula [1] presented in section 2

✓ Calculate the predicted value according to the formula [2] presented in section 2.

✓ If the predicted value is greater than 4, add the movie to the list of suggested movies

✓ If less than 4, increment i to return to the beginning of the loop.

- Provide the list of suggested movies

• The algorithm ends.

Testing algorithm for movie recommendation

Data used for testing: Our testing data is derived from Movies Lens [9] data set. This database includes 100,000 (100k) ratings from 943 users for 1682 movies. We use RMSE to evaluate the model based on the users, which produces a relatively good result approximately 99%.

B. Class diagram of movie recommandation function

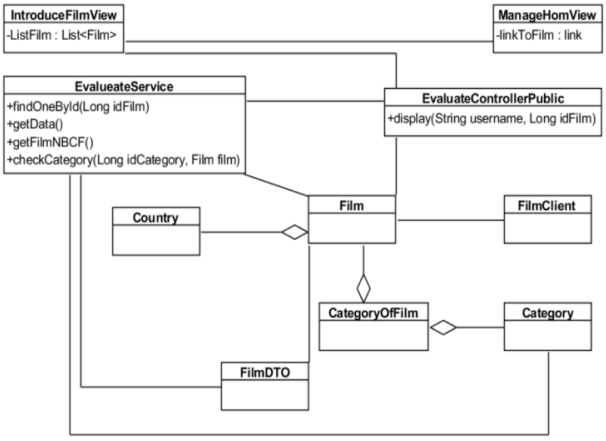

Figure [2]. Class diagram of movie recommendation function

This function needs two interfaces (Figure 2):

- ManageHomeView: user homepage interface
- IntroductFilmView: description of selected movie interface
- Corresponding to 5 procedures at front end :
- display(String username, Long idFilm): get all suggested movies based on username and idFilm
- findOneById (Long idFilm): get movies based on id
- getData (List<Film> film): get data for movie recommendation

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

- getFilmNBCF (int k, Map<String, List<FilmDto>> list_map, String username, List<Film> film): get all suggested movies based on input data
- checkCategory (Long idCategory, Film film): check to see if the suggested movie is of the same genre as the one the viewer is watching.

C. Sequence diagram of movie recommendation function

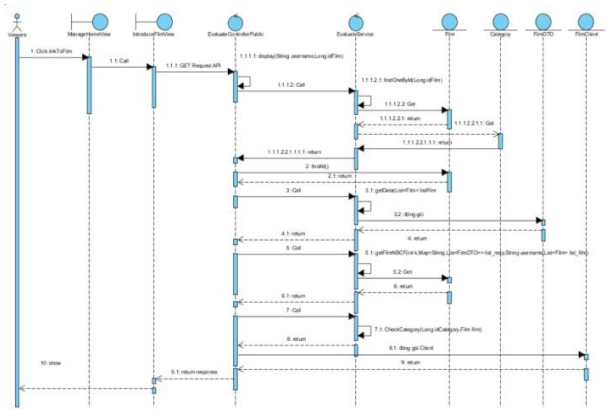

Figure [3]. Sequence diagram of movie recommendation function

IV. CONCLUSION

In this paper, we have carried out extensive study on the NBCF model, the user-based recommendation model. At the same time, incorporating the NBCF model into the developing an online movie streaming website allows users to search for movies of their choice more easily. In this model, we calculate the similarity between two users, thereby making predictions about user reviews with new products. The results show that the suggestions for users are relative, and the system is easy to use, functions well on movie streaming website. The next research direction is that we will develop a larger movie streaming dataset and make some improvements on the algorithm for a better movie recommendation model.

REFERENCES

1. P. Mathew, B. Kuriakose and V. Hegde, "Book Recommendation System through content based and collaborative filtering method," 2016 International Conference on Data Mining and Advanced Computing (SAPIENCE), 2016, pp. 47-52, doi: 10.1109/SAPIENCE.2016.7684166 [CrossRef]
2. Aggarwal, C.C. (2016). Neighborhood-Based Collaborative Filtering. In: Recommender Systems. Springer, Cham. https://doi.org/10.1007/978-3-319-29659-3_2 [CrossRef]
3. Naresh E, Geetha LM, Vijaya Kumar BP; "Recommendation system and its approaches- A survey"; International Journal of Scientific & Engineering Research, Volume 7, Issue 5, May-2016
4. F.O. Isinkaye, Y.O. Folajimi, B.A. Ojokoh; "Recommendation systems: Principles, methods and evaluation"; Egyptian Informatics Journal (2015) 16, 261–273 [CrossRef]
5. A. T¨oscher và M.Jahrer ,Collaborative Filtering Applied toEducational Data Mining , in Proceedings of the KDD Cup2010 Workshop on Improving CognitiveModels with Educational Data Mining,Washington, DC, USA, 2010
6. N. Thai-Nghe, L. Drumond, A. Krohn-Grimberghe, and L. Schmidt-Thieme,"Recommender system for predictingstudent performance," in Proceedings ofthe ACM RecSys 2010 Workshop onRecommender Systems for TechnologyEnhanced Learning (RecSysTEL 2010),vol. 1. Elsevier's ProcediaComputerScience, 2010, pp. 2811 – 2819 [CrossRef]
7. L. Herlocker, J. A. Konstan, et al., AnAlgorithmic Framework for PerformingCollaborative Filtering, Proceedings of the22nd Annual International ACM SIGIR Conference, ACM Press, 1999, pp. 230–237 [CrossRef]
8. U. Shardanand and P. Maes, "Socialinformation filtering: algorithms forautomating 'word of mouth" inProceeding of the SIGCHI Conference onHuman Factors in Computing Systems, ser.CHI '95. New York, NY, USA: ACMPress/Addison-Wesley Publishing Co.,1995, pp. 210–217 [CrossRef]
9. https://grouplens.org/datasets/movielens/100k/

AUTHORS PROFILE

Dr. Phan Thi Ha, is currently a lecturer at the Faculty of Information Technology at Posts and Telecommunications Institute of Technology (PTIT) in Vietnam, and Computing Fundamental Department, FPT University, Hanoi 10000, Vietnam as well. She received a B.Sc.in Math & Informatics, a M.Sc. in Mathematic Guarantee for Computer Systems and a PhD. in Information Systems in 1994, 2000 and 2013, respectively. Her research interests include machine learning, natural language processing and mathematics applications. Email: hapt@ptit.edu.vn , hapt27@fe.edu.vn, hathiphan@yahhoo.com

Trinh Thi Van Anh, is currently a lecturer and a researcher at the Faculty of Information Technology at Posts and Telecommunications Institute of Technology (PTIT) in Vietnam and Computing Fundamental Department, FPT University, Hanoi 10000, Vietnam as well. She received a B.Sc. in Electronics-Telecommunications in 1993 (HUST), M.Sc. in Computer Science in 1998 (HUST). Her research interests include machine learning, NLP and opinion mining. Email: vanh22@yahoo.com, anhttv@ptit.edu.vn, anhttv20@fe.edu.vn