Movie Recommender System

Namratha Shridhar[1], Poorna B S[2], Prakruthi N [3], Preethi S[4], Prof. Pallavi Biradar[5]

[1, 2, 3, 4] School of Computer Science Department, Reva University, Bangalore, India.

[5]Assistant Professor, Reva University, Bangalore, India .

Abstract

A recommender system is always an important factor because of its ability to contribute immense entertainment in our daily lives. It is not only known for entertainment but also for saving our time by recommending exactly what we want. This system can propose movies for the customers depending on their interests or depending on ratings. In this, we propose an engine that suggests movies to existing as well as new users efficiently. When the user attempts to use this system, the main purpose of this will be to mine the information stored, so that an appropriate recommendation can be made. One of the major factors considered while doing this is the previous ratings from the users. Analysis of the ratings with the help of filtering can determine what type of a movie must be recommended to that user. This system mainly focuses on Content and Collaborative Based filtering.

Keywords *- Recommender engine, Content-Based filtering, Collaborative filtering.*

1. INTRODUCTION

A recommender engine is the one that suggests or sorts preferences depending on the customer's interests. These recommender engines are used in a variety of areas like movies, sports, health, and books. Recommendation plays an important role these days because of our busy lives. Searching for something we need becomes infuriating at times. A system that could provide us with suggestions on what we need, without giving any input is highly appreciated. This system suggests depending on the ratings and previously visited genre. If there is a new user without any experience, then the engine suggests depending on the overall rating votes from the existing users. To do this, certain filterings are used:Content-based filtering and Collaborative-based filtering.

Content-based filtering is an AI-applied machine learning method which uses the experience of the user and suggests the movie. This filtering is more often applied in

referral engines, which are algorithms, developed to announce or suggest recommendations to the users. This method of filtering compares the interests of the users with the characteristics of the product. Products whose characteristics are likely to match the most with user interests are recommended. Two methods can be applied here. First, users can receive a list of functionalities from which they can select what they relate with the most. Second, the algorithm keeps in check with the items that the user has already selected and adds these functionalities to the user data profile. Equally, the characteristics of the product can be identified by the product developers.In addition, users may be questioned about what characteristics they feel most closely related to the products. This model is easily scalable because of small quantities of data. In addition, since, unlike other models, it need not compare with the data profile of other users in the engine, it can provide niche specific suggestions to that user. However, this model demands a great deal of knowledge on this filtering method from people who attribute characteristics to the items. Hence, its accuracy and precision are deeply reliable on the extent of knowledge on this method. Furthermore, content-based filtering depends to a large extent on the interests of known users. Therefore, it is limited to the extent that it is incapable of expanding the known interests of users.

Whereas Collaborative filtering uses the similarity between the customers to understand what genre they are looking for. Another feature of this system is to attain the loyalty of the users by providing relevant content and maximizing the time spent by a user on your website. This in turn helps in customer engagement. It runs by going among a big number of people to recognize a selected few people whose preferences are almost similar to those of another user. It examines their most favorite items and generates a list of product recommendations for that user.

The following are the two commonly known subtypes of the collaborative filtering method: User-based: Looks for users who have rated and reviewed products that are almost the same in the same way and then use that rating from the user to get the rating of the movie or item which is missing.

Item-based: In this particular section, we look at the codependency among the two items. With the help of the user's review for the items, this technique locates the items that are missing rating.

In the old days, there were limited options in all the fields. But things are not the same anymore. We need a system that helps us find what we need or at least suggest similar items to what was needed in order to make the best choice and fulfill our requirements.

2. LITERATURE SURVEY

[1] “A movie recommender system: Movrec”, this following paper uses collaborative filtering and provides the user to choose entities on which he/she wants the recommendation, Year:2015, 1 Kumar, 2 M., Yadav, 3 D. K., Singh, 4 A., Gupta V. K.

[2] “Combining content-based and collaborative recommendations” , this system includes two different filtering methods. He also presented an engine that is a product of both, the Bayesian and collaborative technique, Year: 2010 1 De Campos, 2 L. M., 3 Fernández-Luna, J. M.,4 Huete, J. F., 5 Rueda-Morales, M. A.

[3] “Clustering algorithms in hybrid recommender system on movielens data”, analyzes Centroid-based solution and memory based methods, Year:2014, 1 Kuzelewska.

[4] “A survey of recommender systems: approaches and limitations”, analyzes different ways used for suggesting, collaborative, and content-based recommendations. Year: 2013, 1 Sharma, 2 M, 3 Mann, S.

3. METHODOLOGY

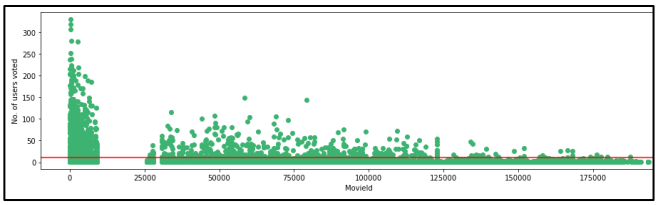
To build a system that works as a recommender the foremost thing that is required is to gather the data which is required in this engine for data analyzation. We gathered movie csv datasets from some popular streaming websites like netflix amazon prime disney hotstar and so on. Along with the movie names we also gathered the genres of those particular movies. Usually the genre of a movie helps us categorize the movies like comedy, action, romance and many more. Each movie can have one or more genres which gives so many different combinations for the user to filter from. The data also includes the existing rating of the movies from well-trusted official movie rating websites such as IMDB and TMDB; it includes the links provided from each movie from well-known streaming websites for users to watch the movie.

The second most important step while building a project is to remove the noise from the gathered dataset to prevent any faulty recommendations. Noise in a dataset is basically missing data i.e., sometimes the dataset might not contain all the data which is required for the system to work smoothly. Removing noise will increase the accuracy and the

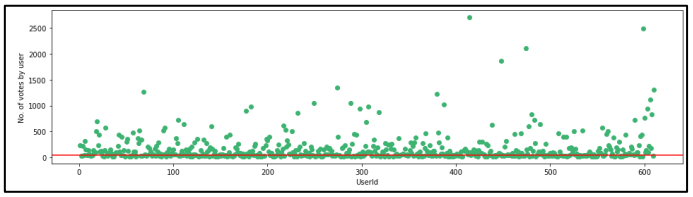
efficiency of the system output. To reduce the noise we conducted some trial and error methods and then filtered out the noisy data.

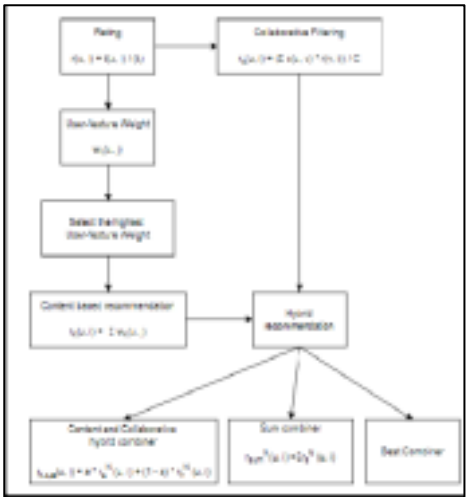
Some of the filters for the csv dataset are:

1. To consider a movie for suggestion at least 10 users should have voted for the movie.



2. To consider a user's vote they should have voted at least 10 movies.





The third step in building this model is removing the sparsity. Sparsity basically happens when a range of values of one or many more dimensions has no data for that specific dataset.

After successfully removing the sparsity of the movie csv dataset, we head on towards creating our movie recommender function. To create this function we use the KNN-algorithm i.e., the k-nearest neighbor algorithm. We have selected this algorithm, as using this can help us generate a movie list which has similar features to the movie entered as input by the user.

The working methodology of this function is rather very simple, first we take the input entered by the user and check if that input is actively there in our movie csv dataset. If the given movie is present in the csv dataset we then use this particular algorithm to find movies that are similar which are present in the csv dataset. The algorithm considers most of the parameters in the movie csv dataset such as the rating genre year of the movie and so on.

```
In [13]: get_movie_recommendation('Iron Man')
```

Out[13]:

	Title	Distance
1	Up (2009)	0.368857
2	Guardians of the Galaxy (2014)	0.368758
3	Watchmen (2009)	0.368558
4	Star Trek (2009)	0.366029
5	Batman Begins (2005)	0.362759
6	Avatar (2009)	0.310893
7	Iron Man 2 (2010)	0.307492
8	WALL·E (2008)	0.298138
9	Dark Knight, The (2008)	0.285835
10	Avengers, The (2012)	0.285319

4. CONCLUSION

In this paper we talk about the movie recommender engine. This system recommends movies depending on the users' likes and dislikes. It takes the data from different users like the ratings and comments on the movie, based on this the recommendation engine suggests a movie that suits more for the user. The system has a dataset of various movies, genres, cast and ratings which is used to help the user to pick out a movie close to their search. The main goal of this paper is to provide a time saving and good movie recommendation for the user and not disappoint them. Users must feel the ease

of using the system and get a good output from it. By recommending a movie faster the user is being entertained and can have a good experience.

ACKNOWLEDGEMENT

We are glad to have this project completed successfully. It is very important to acknowledge everyone who has helped us complete this project.

Firstly we are thankful to our guide Prof. Pallavi Biradar, Reva University for her guidance in this project.

Secondly we express our gratitude to the School of Computers and Information Technology for their co-operation in this project.

Lastly we are thankful to the project coordinator prof raghavendra reddy for his support in coordinating everything regarding the project.

REFERENCES

[1] Kumar, M., Yadav, D. K., Singh, A., & Gupta, V. K, "A movie recommender system: Movrec", International journal of computer applications, 124(3), 7-11, (2015).

[2] De Campos, L. M., Fernández-Luna, J. M., Huete, J. F., & Rueda-Morales, M. A. "Combining content-based and collaborative recommendations: a hybrid approach based on Bayesian networks", International Journal of approximate reasoning, 51(7), 785-799,(2010).

[3] Kuzelewska, U," Clustering algorithms in hybrid recommender system on movielens data", Studies in logic, grammar and rhetoric, 37(1), 125-139,(2014).

[4] Chiru, C. G., Preda, C., Dinu, V. N., & Macri, M," Movie recommender system using the user's psychological profile", 2015 IEEE international conference on intelligent computer communication and processing (ICCP) (pp. 93-99). IEEE, (2015, September).

[5] Sharma, M., & Mann, S," A survey of recommender systems: approaches and limitations", International Journal of Innovations in engineering and technology, 2(2), 8-14, (2013).

[6] Li, P., & Yamada, S," A movie recommender system based on inductive learning",

IEEE conference on cybernetics and intelligent systems (pp. 318-323). Singapore, Singapore: IEEE, (2004, December)

[7] Vilakone, P., Park, DS., Xinchang, K. et al. An Efficient movie recommendation algorithm based on improved k-clique. Hum. Cent. Comput. Inf. Sci. 8, 38 (2018)

[8] A. Jain and S. K. Vishwakarma, "Collaborative Filtering for Movie Recommendation using RapidMiner", International Journal of Computer Applications, vol. 169, no. 6, pp. 0975-8887, July 2017.

[9] A.V. Dev and A. Mohan, "Recommendation system for big data applications based on set similarity of user preferences", 2016 International Conference on Next Generation Intelligent Systems (ICNGIS), pp. 1-6, 2016.

[10] H. W. Chen, Y. L. Wu, M. K. Hor and C. Y. Tang, "Fully content-based movie recommender system with feature extraction using neural network", 2017 International Conference on Machine Learning and Cybernetics (ICMLC), pp. 504-509, 2017.

[11] Basha, Syed Muzamil, and Dharmendra Singh Rajput. "An innovative topic-based customer complaints sentiment classification system." *International Journal of Business Innovation and Research* 20, no. 3 (2019): 375-391.