Movie Recommender System Using Machine Learning Algorithms

Piyush Kumar¹, Shaik Golam Kibriya¹, Yuva Ajay¹, Ilampiray¹
¹Dept. of Information Technology, Sri Krishna College of Technology, Coimbatore, India
17tuit102@skct.edu.in

Abstract. These days, a recommendation of a movie from a server-based system has made finding a piece of cinema easier. Film recommendation helps us to find films that we need to watch, instead of searching extensively online and help cinephiles and movie buffs by suggesting top tier films to watch without looking into huge databases which is very time consuming. As an approach to this dilemma, we introduce a model based on collaborative and content-based approach which will use a variety of Python based Machine Learning algorithms from huge datasets and immensely produce a movie suggestion based on their taste and past watch history or genre. This compared to other recommendation systems is different and is based on a content-based approach.

1. Introduction
The release of thousands of movies all across the world has been increasing day-by-day across all platforms and countries. Accessing desirable content to watch these movies has been a part of life since the beginning of the entertainment business. To find a suitable movie to watch with family, or to watch in your free time becomes a crucial part of the decision-making process. Solving these problems requires an intelligent system known as Recommender systems which allows the client to choose a movie of their taste, from a dataset of thousands of movie databases [1-4].

A recommender system is profitable for both the client and the service provider by recommending accurate and obvious suggestions to the client to watch a movie of their previous taste or selection. A recommendation system successfully recommends various kinds of media such as Music, Movies, Adverts, Items in grocery Store, News and all sorts of things which have Options. They use different kinds of methods and algorithms to rank or grade these movies, majorly based on the percentage of users who have been satisfied by the purchase of a product, and further recommend them according to their ranking [5-8]. Which in turn results in Decreasing the search time of the user, and increasing the probability of purchase by the customers, and retaining customers who were satisfied for further services are some of the advantages for having a recommender system.

According to a survey, users choose to invest on sites which utilize recommender systems and users in the best and luxury contributing category to uphold an efficient recommendation system. Generally, the learning system learns from different data, such as a) User's action and client's preference and dislikes, b) Reviews and feedback from such customers to recommend relevant films which they bought.
Recognizing and recommending media which are to be provided to clients are usually similar to the client’s watch or purchase history, which is the main part of the recommendation system. But the generalization used by the recommender system is another important aspect [9-14]. Unfortunately, many clients do not participate in feedback for rating movies while they implicitly show their opinion everywhere else, i.e., Users can rate movies on the platform when they buy or watch them, which also increases the accuracy of the system. The outcome of using the proposed methodology shows better and improved performance than an arbitrary recommender system. From a Company’s point of view, the more relevant products a client finds on a given platform, the higher their engagement rate is predicted to be. This often results in increased revenue for the platform itself and they are able to retain their customers too.

2. Related Works
The analysis on recommendation of movies was initiated during early 2000's with the increase in streaming, shopping platforms such as Netflix and Amazon. The analysis in the streaming platforms many approaches are arranged, the approaches are divided into clusters: Content-based, hybrid and collaborative filtering. These are some of the methods which can be used to recommender systems.

Content-based filtering computes similarity between clients’ likes built on few formulae and offers media or things which are identical to a distinct customer’s likes and their feedback.

There is another filtering method known as Collaborative filtering generated by comparing and scoring the likes and dislikes of a user by collecting their tastes along with the same information from several other clients with an akin and similar purchase history. It works extremely well when you are provided with a huge sample of clients as compared for a comparatively less group of clients with comparable history [15-18]. Building a creative and capable movie recommendation system which can grasp to deliver enhanced guidance as higher data about clients are assembled.

These methods suffer drawbacks too. Data needs and train-test data restriction are some of the flaws for these processes.

3. Data
The set of data that will be used in applying the algorithms, consists of 2,60,00,000 ratings and 7,50,000 tags applied to 45,000 movies by 2,70,000 clients. This also includes tags with 12 million relevancy score across 1,100 different tags.

4. Methodology
As mentioned in the Introduction, the proportion of view is the main benchmark we are going to use for the recommendation of movies. To conclude this benchmark, content-based filtering algorithms, collaborative filtering with a new methodology known as the "residual method" will also be used.

5. Data Visualisation
The Figure below shows the graph of percentage of movie watched by the total number of users for all the movies in the dataset.
Figure 1. Data Visualisation of the spike of views depicted in a histogram

The span of various views on the website is less than 10.63% of the duration of the films. The characteristic between client’s behaviours and manners in viewing or leaving the movie at the half-time or interval, creates two distinct spikes in the two points of the histogram graph. The percentage of view is hence hindered to 89.27% to exclude the bias of viewing the end-credits between clients Figure 1.

6. Like-View Correlation

The percentage of films watched or admiring a movie, by visualizing the relation among them, the sights of these proportions is allotted into periodic positions. The proportion of the total number of likes the movie has received is considered along with the number of viewers in each period. There is a relationship between the like and percentage of the movie viewed by the clients and perhaps a positive correlation between the two entity. As this shows, the dimension differs in the value of probability of disapproval of the movie watched by the user is around 0.02 percent, which is comparatively minute when compared to the movies which left the users satisfied. The ambiguity of fitting the curve into the graph is comparably immense Figure 2.

Figure 2. Data-visualisation of relation between liking a movie and the percentage of movie watched by the user shows a linear relation with confidence of 95%
In both of the above figures, the percentage of movie viewed is limited to 90% since almost all the viewers usually skip the opening and end credits which are presented at the beginning and end of every movie Figure 3.

7. Prediction of Percentage Of View
In this part of the paper, different percentages of view by different learning algorithms have been explained briefly and the evaluated results and accuracy of these methods have been evaluated using certain evaluation methods.

A. Content-based Strategy
Content-based strategies utilize the movie highlights and other meta-data along with the client's watchhistory to separate the client's inclination towards a movie. In this paper, movies attributes are film type category, release date, IMDB ratings, actors, co-actors, directors, screenplay writer, cinematographer, background sound etc alongside the mean, median and mode of the level of time of movie-view and number of perspectives on the film. Each one of these highlights is encoded utilizing one-hot encoding technique by sci-kit learn realize which encode writings to 0 and 1. Also, the highlights that are utilized for clients are the number of motion pictures watched by the client, for eachgenre, for every film type, average of the level of view, median of the level of the view and sexual category. Linear regression with multiple variables and Random forest regression are used as the algorithms to predict movies in this method which give us a good accuracy over other Machine Learning Models.

B. Collaborative Filtering
Collaborative filtering, is an algorithmic rule creating automatic predictions by filtering the interests of a movie watcher by aggregation of their preferences or style data from many purchasers with similar getting and action history. A well-liked filtering methodology for this method is factorization of the matrix of a given dataset. This approach factorizes user - movie rate lattice into user grid and movie matrix. The considering streamlines a misfortune work over the user-movie grid. The dropping error function is shown in (1) in which $A$ is the client, $B$ is denoted as movie, $a_0$ is row of the user matrix, $b_1$ is the column segment of the film matrix, $U$ is the quantity of clients in the information base. Movie is the total number of movies present in the dataset, $W$ is the weight limitation, $\lambda$ is known as the standardization factor and $R (A, B)$ is a standardised term. This
The condition implies the calculation limits

of the root of mean squared error of forecast over the grading which are now known in the client movie rate matrix.

\[ \text{argmin}_{a,b} \sum \sum W_{ui} (r_{ui} - a_{u} b_{i})^{2} + \lambda R(A,B) \] (1)

Using the above formula (1), the client-movie matrix with mentioned size of U * Mov is factored into client lattice with size of U * K and movies matrix with size of K * Mov where K is the number of meta-data such as features and is chosen lower than U and Mov. So, the number of parameters that should be estimated to perform this filtering through this new model is equal to U * K + K * Mov. It is not exactly the number of void parts inside the client thing rate network and as a result, it will alter the pace of assessment.

C. Collaborative Filtering based on Memory

This strategy which is a closeness-based technique and comparable clients/things will be discovered utilizing the client-movie matrix along with KNN classifier algorithm. At that point, the level of length of movie viewed can be assessed by clients with similar tastes for some movies. The strategy that finds comparable clients is known as the client-based method and the strategy that finds comparative properties across the dataset is known as the movies-based strategy. Since the client based and movie-based strategies are fundamentally the same, in this segment, just the movie-based technique is clarified. The two popular movies, named as “Inception” and “Tenet” which are very much comparable and similar are watched by the client 1 and client 2. In this way, since client 3 has watched about 75% of the movie “The Prestige”, the presented calculation predicts that he/she would watch 75% of the movie “Tenet” as well. In this paper, the Mean Square Distance of the prediction is the method used as the metric of similarity Figure 4.

| INCEPTION | TENET | THE PRESTIGE | USER |
|-----------|-------|-------------|------|
| 90        | 90    | 70          | 2    |
| 100       | 90    | 75          | 3    |

Figure 4. client-movies matrix for collaborative filtering.

D. Residual Based Algorithm

The residual method is a general construct, the thought is to apply a method or algorithm in such a way that can try a basic prediction of data and then apply those results to a second model and evaluate the predictions with the first layer of the ensemble mode. This will be applied to multiple and successive levels of predictions and will further deepen and create a strong model. The ultimate aim of the prediction tactic is that the summation of the primary prediction with all of the anticipated errors in the alternative levels. This methodology may be a sensible approach to cut back the ultimate error however there's a sensible downside with it because the depth of the model will increase on every prediction level and results in decrease of absolute value and mean of absolute value of the error which results in high bias in the model and the model cannot find the percentage of error in the model. To boost the accuracy of the prediction, the sign and therefore the value of error will be calculable every level.

8. Evaluation Metric
For successful and accurate evaluation of the model, the mean of absolute error method is selected as the metric for evaluation, as shown in (2)

\[
MAE = \text{absolute (real value – predicted value)} \quad (2)
\]

| Algorithm      | Method        | MAE  |
|----------------|---------------|------|
| Linear Regression | Content-based | 26.9%|
| Random Forest  | Content-based | 26.5%|
| Model-based    | Collaborative | 27.23%|
| Client-based   | Collaborative | 25.8%|

### 9. Recommender System

The total watch time of all the listed movies in the set can be guessed by predicting it for a specific user, and by principle the movies with those of highest watch time of the predicted view or similarity score will be suggested to the user. The data will not be split randomly for evaluation. The splitting of data is done into 80%-20% split. In which 80% is considered as the training data for the client and the remaining 20% will be considered as the assessment data. The precision of this system can be determined by taking the proportion of matching movies from the recommended list of movies and comparing them with the test list of the users who have watched similar movies to the size of the suggested list created for the user. For comparison, a random recommender system is considered as the baseline to evaluate our current recommender system. Different methodology and algorithms have been successfully tested on this dataset and our system recommends movies whose performance is better by 3 times than a randomly built recommender system.

### 10. Conclusion

We have applied both content-based and collaborative filtering based on the meta-data of the movie to construct the movie recommendation system. In Collaborative filtering algorithms the overall performance is better than the content-based filtering in terms of test Root Mean Square system. Content based filtering is computationally costlier than collaborative filtering, as it involves huge processing of text features. Hence collaborative filtering is preferred systems work on separate client’s ratings, therefore limiting your choice to explore more. While these systems, which are based on a collaborative approach, figure the connection between different clients and depend upon their ratings, prescribe movies to others who have similar tastes, allowing clients to explore a lot. It also allows clients to rate movies as well and recommends them suitable movies based on other’s ratings.

### References

[1] G. Popescu, *Group recommender systems as a voting problem*, in Online Communities and Social Computing, ser. Lecture Notes in Computer Science, A.Ozok and P. Zaphiris, Eds. Springer Berlin Heidelberg, vol. 8029, pp. 412–421, 2013.

[2] Jain, A., & Vishwakarma, S. K., *Collaborative Filtering for Movie Recommendation using RapidMiner* International Journal of Computer Applications (0975 - 8887) Vol. 169 - No. 6, 2017.

[3] J. Sun, G. Wang, X. Cheng, and Y. Fu, *Mining affective text to improve social media movies recommendation*, Information Processing & Management, vol. 51, no. 4, pp. 444–457, 2015.

[4] K. Soni, R. Goyal, B. Vadera, and S. More, *A Three Way Hybrid Movie Recommendation*
System, International Journal of Computer Applications, vol. 160, no. 9, pp. 29–32, 2017.

[5] H. Li, J. Cui, B. Shen, and J. Ma, An intelligent movie recommendation system through group-level sentiment analysis in microblogs, Neurocomputing, vol. 210, pp. 164–173, 2016.

[6] F. Wu, Y. Huang, Y. Song, and S. Liu, Towards building a high-quality microblog-specific Chinese sentiment lexicon, Decision Support Systems, vol. 87, pp. 39–49, 2016.

[7] Y. Wang and W. Xu, Leveraging deep learning with LDA-based text analytics to detect automobile insurance fraud, Decision Support Systems, vol. 105, pp. 87–95, 2018.

[8] D. Devikanniga, A. Ramu, and A. Haldorai, Efficient Diagnosis of Liver Disease using Support Vector Machine Optimized with Crows Search Algorithm, EAI Endorsed Transactions on Energy Web, p. 164177, Jul. 2018. doi:10.4108/eai.13-7-2018.164177

[9] H. Anandakumar and K. Umamaheswari, Supervised machine learning techniques in cognitive radio networks during cooperative spectrum handovers, Cluster Computing, vol. 20, no. 2, pp. 1505–1515, Mar. 2017.

[10] Mohamed Sarwat, Justin J Levandoski, Ahmed Eldawy, and Mohamed F Mokbel. Lars An efficient and scalable location-aware recommender system. IEEE Transactions on Knowledge and Data Engineering, 26(6):1384–1399, 2014.

[11] M. Deshpande and G. Karypis, Movies-based top- N recommendation algorithms, ACM Trans. Inf. Syst. vol. 22, no.1, pp. 143–177, 2004.

[12] Manoj Kumar, D.K. Yadav, Ankur Singh and Vijay Kr. Gupta (2015), A Movie Recommender System: MOVREC, International Journal of Computer Applications (0975 – 8887) vol. 124 – No. 3.

[13] Linyu Sua and Xuebin Chen, Improvement of client-based collaborative filtering algorithm, J.Computer engineering and software, vol. 38, pp. 127-132, 2017.

[14] Y. Xu, X. Guo, J. Hao, J. Ma, R. Y. K. Lau, and W. Xu, Combining social network and semantic concept analysis for personalized academic researcher recommendation, Decision Support Systems, vol. 54, no. 1, pp. 564–573, 2012.

[15] Y. Zhang, D. Zhang, M. M. Hassan, A. Alamri, and L. Peng, CADRE: Cloud-Assisted Drug Recommendation Service for Online Pharmacies, Mobile Networks and Applications, vol. 20, no. 3, pp. 348–355, 2015.

[16] J. R. Priester and R. E. Petty, The Gradual Threshold Model of Ambivalence: Relating the Positive and Negative Bases of Attitudes to Subjective Ambivalence, Journal of Personality and Social Psychology, vol. 71, no. 3, pp. 431–449, 1996.

[17] J. Sun, W. Xu, J. Ma, and J. Sun, Leverage R&F to find domain experts on research social network services: A big data analytics methodology with MapReduce framework, InternationalJournal of Production Economics, vol. 165, pp. 185–193, 2015.

[18] Y. Koren, R. Bell and C. Volinsky. Matrix Factorization Techniques for Recommender Systems, Computer, vol. 42, n. 8, p. 30-37, 2009.