IFERP
connecting engineers... developing research

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

MOVIEREC: A Case study on Movie based Recommender

[1] M.Ravi Teja , [2] Amarendhar Reddy
[1][2] Anurag Group of Institutions.

Abstract: MOVIEREC was created for moviegoers who want to find movies of their taste. A moviegoer may have to spend a lot of time on the web reading and watching reviews to reach a conclusion as to whether he/she should watch a movie or not. Usually the information available in various media about the movies are not targeted towards user's taste , or may display the information which may not always stay on the topic and slowly drift away from the viewer's interests. MOVIEREC involves taking the user's feedback and finding similarity of the user to other users by using collaborative filtering and clustering algorithms. These algorithms, based on the user's feedback, place the user into a particular cluster and recommend the user the movies that he/she may like based on the feedback given by users who fall in the same cluster. MOVIEREC application aims at reducing the time spent in searching for a movie of his/her taste and thereby increasing chances of getting value for their movies.

Keywords: Collaborative Filtering , Content-based Recommender system , Expert Users ,Cluster Formation ,K-Means

1. INTRODUCTION

MOVIEREC is a web application in which a user gives ratings to movies and builds a custom taste profile. The system then uses collaborative filtering and clustering algorithms to find similar movies based on the custom taste profile and recommends movies to the user. This learns continuously user preferences, and keeps getting better with usage over time. The web application is built using Bootstrap which is a responsive mobile first framework.

Movie Recommender targets all people who are interested in movies, but find it difficult to find movies which are to their taste. The application is designed in such a way that the user initially rates a few movies which he has seen and then the application builds a custom taste profile and recommends movies which are closer to his taste profile. The user can then view the list of recommended movies and watch them. The recommendations keep getting better with each rating given by the user.

In the existing system, people had to ask their friends or visit sites like IMDB or YouTube, and he has to read a lot of reviews and then decide on a movie to watch and even then there is no guarantee that the user may find a movie that he may like. People usually end up with watching movies that they are not satisfied with. We overcame these drawbacks in the proposed system.

II. RELATED WORK

Recommender systems are a type of information filtering system that gives advice on products, information, or services that a user may be interested in. They assist users with the decision making process when choosing items with multiple alternatives. Recommender systems are popular due to their e-commerce application purposes. Within the e-commerce world, recommendations provide aid to customers and help them find what they may be looking for, thus increasing business. In addition, it can be used as a tool to predict user's behavior, but should not be used to select recommendations on their behalf.

There are two basic entities that appear in any recommender system are the user and the item of interest. The user can be a customer in an e-commerce platform or an avid book reader looking for a recommendation for the next book they should read. The users provide their ratings on items and are used to aid other users with their recommendations. The item is the second piece of a recommender system. Users give items ratings and the algorithm outputs recommended items based on new user queries.

We can classify the recommender systems in three broad categories:
i. Collaborative Filtering Recommender system
ii. Content-Based Recommender system
iii. Demographic based Recommender system
iv. Hybrid Recommender system

All Rights Reserved © 2018 IJERCSE

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

I. Collaborative Filtering

Collaborative filtering, the traditional Recommender system is based on rating structure usually represented as User-Movie Rating matrix. Each cell value represents the rating of a movie by the user. It predicts the ratings based on similarity measures like Pearson correlation coefficient, cosine similarity, Euclidean distance measure, etc. Collaborative filtering can be classified into two types: Memory based CF and Model based CF. Memory based CF predicts the ratings using the entire user-item database of users who are similar to the active user whereas Model based CF predicts the ratings by using the constructed model. Collaborative filtering often suffers from several issues which include

Sparsity: Most often users do not rate the movies which results in sparsity of data.

Cold start: To recommend a new item or for new user who has not yet rated any movies, it is very difficult as there exists no user information.

Scalability: To handle millions of users and movies over Internet, CF computations to find similar users grow exponentially and becomes expensive.

ii. Content Based Recommender System.

Content based recommendations are based on the user individual preferences and tastes. It recommends the movies preferred by the user in the past.

Content-based Recommender system often suffer from the following issues:

Limited content analysis It is difficult to recommend if there is a limited content about the user profile Overspecialization restricts users to items similar to the ones defined in their respective profiles and thus new items and other options are not discovered.

iii. Demographic Recommender system.

Demographic Recommender system generate recommendations based on the user demographic attributes. It categorize the users based on their attributes and recommends the movies by utilizing their demographic data . In contrast to collaborative filtering and content based recommender system, it is easy to implement and does not require user ratings.

iv. Hybrid Recommender system.

Hybridization of demographic and collaborative filtering approaches had been employed to solve cold start problem. Hybrid model based approach has been applied on Movie data set to enhance recommendation quality. Additionally Collaborative filtering and Demographic based approach had been used to modify similarity calculation.

In contrast, this paper proposes a novel approach to enhance the Recommendation quality by utilizing user demographic data provided by the users explicitly and Collaborative filtering approach based on user ratings on movies.

III. PROPOSED METHODOLOGY

In this proposed system user overcomes the disadvantages of the existing system by using MOVREC web application. If the user wants to watch a movie and doesn't know which movie to watch he registers himself with the application and rates the movies he has already watched and builds a custom taste profile, and depending on the ratings, he is recommended movies by the application.

The methods that we have used in this Proposed system are

a. Generation of Expert Users
b. Cluster Formation
c. Recommendation generation

a. Generation of Expert Users:

The algorithm initially splits the users into groups by filtering based on demographic data such as marital status, gender, etc. Then we calculate the average ratings for each movie belonging to same genre i.e. action, comedy, romantic, horror, etc., then we find how similar are the users belonging to the filtered group are to the average values of movies by using Euclidean distance formula. Then the user who is the most similar to the average ratings is set as the expert user for that genre for that filtered group.

b. Cluster Formation

We take the ratings for the movies that are given by new user and expert users as the inputs and perform K-means clustering to form clusters and to know under which cluster the new user falls into.

Algorithm:

i. Let X = {x1,x2,x3,……..,xn} be the set of data points and V = {v1,v2,…….,vc} be the set of centers.

ii. Randomly select 'c' cluster centers.

All Rights Reserved © 2018 IJERCSE

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

iii. Calculate the distance between each data point and cluster centers.
iv. Assign the data point to the cluster center whose distance from the cluster center is minimum of all the cluster centers.
v. Recalculate the new cluster center using:

$$V_i = (1/C_i)\sum_{j=1}^{C_i} X_i \quad (1)$$

where, 'ci' represents the number of data points in ith cluster.
vi. Recalculate the distance between each data point and new obtained cluster centers.
vii. If no data point was reassigned then stop, otherwise repeat from step iii.

c. Recommendation generation

Generating a neighbourhood involved calculating the similarity between the given users within the user-item matrix. Similarity will be used to generate a recommendation for a specific user.
Algorithm:
i. Compare the similarity between all users with the active user.
ii. Select n users that have the highest similarity to build a neighbourhood
iii. Compute the prediction based on this similarity matrix.

We compute similarity between users by finding the Euclidian distance between them. The similarity between users x1 and x2 is calculated below:

$r_2(x^1, x^2) = Dist(x^1,x^2) =$

$$\sqrt{(x_1^1 - x_1^2)^2 + (x_2^1 - x_2^2)^2 + \cdots + (x_D^1 - x_D^2)^2} = \sqrt{\sum_{d=1}^{D}(x_d^1 - x_d^2)^2} \quad (2)$$

From equation (2) We compute the similarity between two users x and y by performing the following calculations

$$sim(x,y) = \frac{1}{1+r_2(x,y)} \quad (3)$$

We recommend movies to user by weighting the ratings of each "Expert User" by the similarity to user. An overall score for each film is obtained by summing these weighted scores. If u is the user and we have C expert users, then the estimated score given to a movie z by user u, scu(z) is obtained as follows:

$$sc_u(z) = \frac{1}{\sum_{c=1}^{C} sim(x^u,x^c)} \sum_{c=1}^{C} sim(x^u,x^c) \cdot sc_c(z) \quad (4)$$

2. The Basic K-means Algorithm

K-means Clustering is an unsupervised learning which is used when we have unclassified data. The goal of this algorithm is to classify a given data set into different number of clusters . K-means algorithm works iteratively to give each data set point of k different groups based on the specific features.

$$J(V) = \sum_{i=1}^{c}\sum_{j=1}^{c_i}(\|x_i - v_j\|)^2 \quad (5)$$

where, '||xi - vj||' is the Euclidean distance between xi and vj. 'ci' is the number of data points in ith cluster. 'c' is the number of cluster centers.

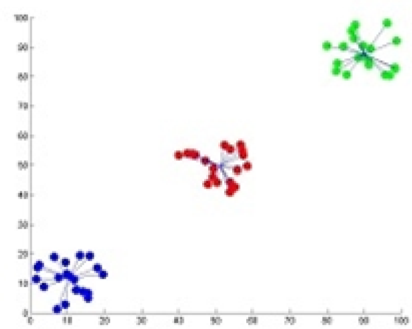

Fig 1:Showing the Result of K-Means

3. Data Description

In this proposed system user overcomes the disadvantages of the existing system by using MOVREC web application. If the user wants to watch a movie and doesn't know which movie to watch he registers himself with the application and rates the movies he has already watched and builds a custom taste profile, and depending on the ratings, he is recommended movies by the application.
In the proposed Model the attributes used to calculate distance of each point from the centroid are
i. Genre
ii. Gender
iii. Marital status

All Rights Reserved © 2018 IJERCSE

IFERP
connecting engineers... developing research

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

iv. Ratings
Here, different attributes have different weights and In our Recommended System.
Every time when the user gives a rating to a particular movie an Expert for that Cluster in which that user belongs to will change dynamically.

CONCLUSION

"Movie Recommender" web based application. This application helps the user to find movies which are similar to his/hers taste. The existing system requires the user to go through several websites and read and watch reviews to come to a conclusion which is not efficient. The intension of this application is to reduce the time spent and increase efficiency by using collaborative filtering techniques.

REFERENCES

[1] Kantor, P. B., Rokach, L., Ricci, F., & Shapira, B. (2011). Recommender systems handbook. Springer.

[2] J.B.Schafer,D.Frankowski,J.Herlocker,S.Sen, Collaborative filtering recommender systems. The Adaptive Web, 291-324, 2007

[3] G. Adomavicius, A. Tuzhilin, Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Transactions on Knowledge and Data Engineering, 17(6), 734-749, 2005.

[4] M.J. Pazzani, A Framework for Collaborative, Content-based and Demographic Filtering. Artificial Intelligence Review, 13(5-6), 393-408, 1999

[5] Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A. (2013). Recommender systems survey. KnowledgeBased Systems, 46, 109-132.

[6] Han, J., Kamber, M., & Pei, J. (2011). Data mining: concepts and techniques: concepts and techniques. Elsevier

[7] M. O. Conner and J. Herlocker. Clustering Items for Collaborative Filtering. In Proceedings of the ACM SIGIR Workshop on Recommender Systems, Berkeley, CA, August 1999.

[8] S.H.S. Chee, J Han, K. Wang. Rectree: An efficient collaborative filtering method. Lecture Notes in Computer Science, 2114, 2001

[9] L. H. Ungar and D. P. Foster. Clustering Methods for Collaborative Filtering. In Proc. Workshop on Recommendation Systems at the 15th National Conf. on Artificial Intelligence. Menlo Park, CA: AAAI Press.1998

[10] Yuanyuan Wang, Stephen Chi-fai Chan and Grace Ngai Applicability of Demographic Recommender System to Tourist Attractions : A Case Study on Trip Advisor , 2012 IEEE DOI 10.1109/WI-IAT.2012.133

[11] B. Sarwar, G. Karypis, J. Konstan and J. Riedl, Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering, Proceedings of the Fifth International Conference on Computer and Information Technology, 2002

[12] A. Said, T. Plumbaum, W. E. De Luca, and S. Albayrak, "A comparison of how demographic data affects recommendation," in Proc. 19th international conference on User modeling, adaption, and personalization, 2011.

All Rights Reserved © 2018 IJERCSE