International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 07 | July 2020 www.irjet.net p-ISSN: 2395-0072

Movie Recommendation System

Ananya Agarwal[1], S. Srinivasan[2]

[1]*Student, Dept. of Computer Science and Engineering, Galgotias University, Uttar Pradesh, India*
[2]*Professor, Dept. of Computer Science and Engineering, Galgotias University, Uttar Pradesh, India*

Abstract - *Filtering systems are often used to remove unnecessary information from a large amount of data. Recommender systems are used to seek and predict meaningful and informative items that a user might put into the data. The system emphasizes on reusing the information and preferences of the users that can be used in the calculation of future recommendations. This paper proposes a recommender system which provides recommendation based on the information given by the users. It is done by using analysis of user's psychological profile, their watching history and movie scores from other websites. It is actually based on aggregate similarity conditions. This system uses both content and collaborative filtering. Both can be explained as follows: Collaborative filtering means building systems from user's past behavior (ie. Items that have already been selected or rated) Afterwards the model is used to predict outcomes that the user might be interested in.*

Content based filtering uses a series of distinct and discrete characteristics of an item in order to recommend more items with same properties.

Both of these systems combine to make a hybrid recommender system.

This system which is a hybrid of both filtering systems is capable of recommending movies using analysis of the profiles.

Key Words: Collaborative Filtering, Content based Filtering, Hybrid Filtering, Hadoop, K-means Algorithm.

1. INTRODUCTION

A recommendation system is a model which is used to filter information and predict the output based on the preferences of the user. These models have become extremely popular that they are being used in movies, books, television, restaurants, food etc. These systems help in improving the future suggestion of the company.

A large number of companies are benefiting from the recommendation system in improving customer satisfaction and experience. In this way they are collecting massive chunks of revenue which is why most of them are turning to a recommendation system.

One goal of this system is to provide a system that considers the past ratings given by the user to provide suggestions to the user. It is implemented using the collaborative filtering and Apache Mahout framework. The second goal is to compare the performance and efficiency of user-based recommender system and item-based recommender system.

1.1 OVERVIEW:

Collaborative filtering: It means building systems from user's past behaviour. Afterwards the model is used to predict outcomes that the user might be interested in.
Content based filtering: It uses a series of distinct and discrete characteristics of an item in order to recommend more items with same properties.
Hybrid System: Hybrid recommendation systems are a combination of both collaborative and content-based filtering methods. In these type of systems, collaborative and content-based predictions are performed separately and then the results of both techniques are combined to provide recommendations.
Mahout: It aims to produce free and distributed and scalable implementations of advanced machine learning algorithms used in the field of clustering, classification, collaborative filtering and frequent pattern matching it offers a full stack option of incorporating machine learning on the big data which is managed by the underlying Hadoop platform.

1.2 EXISTING MODELS

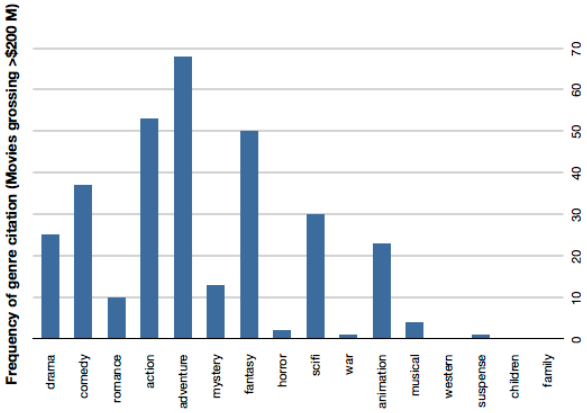
Over the years, many recommendation systems have been developed using either collaborative, content based or hybrid filtering methods. These systems have been implemented using various big data and machine learning algorithms.

The authors propose a collaborative recommendation system which is designed to work on the Hadoop platform, using the Map Reduce framework. The authors have used the set-similarity join method to build this system, employing both user-based and item-based collaborative filtering techniques.

They proposed a movie recommendation system using collaborative filtering that focuses on the ratings given by the users to provide recommendations. The proposed system is built using K-means algorithm to sort the movies according to the ratings.

In one paper the authors propose a fully content-based movie recommendation system to recommend movies. The proposed system makes use of a neural network with the

content information of the movies to obtain features and learn the similarities between movies.



The movies are recommended based on the similarity between them.

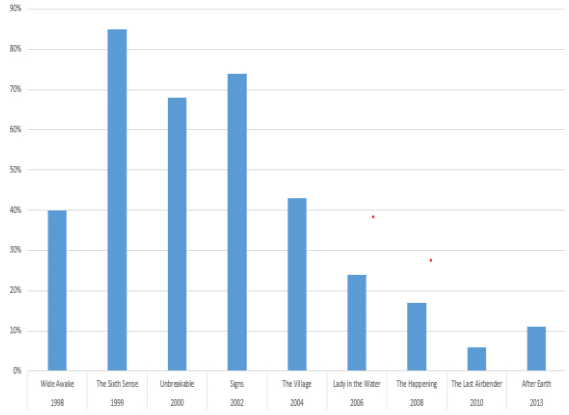
The authors implement a recommendation system that combines both user-based and item-based collaborative filtering approach. The system is built using nearest neighbours machine learning technique and develop a new algorithm that unifies used based and item-based recommendations. Based on the research we conducted, collaborative filtering was found to be one of the popularly used approaches to build recommendation systems. Many of the systems used machine learning algorithms such as clustering using K-means, neural networks and so on to recommend items.

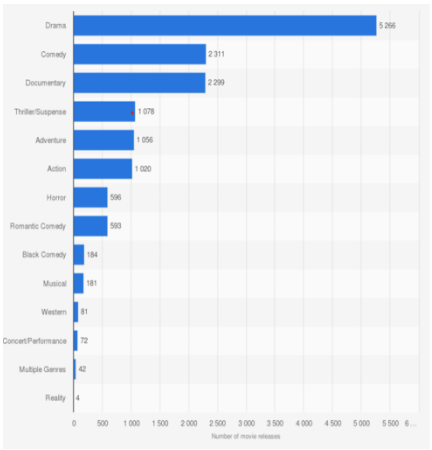
2. IMPLEMENTATION

1. Dataset: The dataset used in this paper is obtained from Yahoo Research Webscope database. It provides two files - Yahoo! Movies User Ratings and Yahoo! Descriptive Content Information. The Yahoo! Movies Users Ratings file contains 211231 records and consist of User ID, Movie ID and Ratings. The Yahoo! Movies Descriptive Content Information file contains 54058 records and consists Movie ID, Title, Genre, Directors, Actors and etc.

2. Data Cleaning: The Movies Descriptive Content Information file contained about 40 columns. Most of these columns were not required for our experiments and hence were removed. The dataset also contained a lot of blank values and duplicate values which needed to be resolved. In addition, there were some entries for movies in the Movies Users Ratings files that didn't correspond to any movie in the Movies Descriptive Content Information file. These entries were removed for easier processing.

3. Data Analysis: We analysed and gained insight that could help in developing our system using the Matplotlib libraries in python. The patterns are identified such as most rated movies, most rated genres most rated genres, the number of movies in each genre and the number of movies rated in each rating category.





4. Model building: We used the mahout library to build the recommender system. For User-based filtering we used the User-Similarity class in addition to the Pearson Correlation Similarity which uses the Pearson Correlation Coefficient to determine the similarity between users' ratings. Below is the mathematical formula for Pearson Correlation. The higher the coefficient is, the more correlated the two users' choices are.

$$r = \frac{\sum(x-\bar{x})(y-\bar{y})}{\sqrt{\sum(x-\bar{x})^2 \sum(y-\bar{y})^2}}$$

The User-neighbour-hood is computed by using another machine learning algorithm of distance-based clustering called Nearest N User Neighbour-hood where N is defined in the program code. Nearest Neighbour algorithm searches the N nearest data-points around each data-point to get the most similar data-points and group them together. The item-based filtering is implemented using the Item Similarity class of Mahout. The machine learning algorithm used to compute the item similarity is Log Likelihood Similarity. Since the

items are static, their similarities based on the user ratings are not going to change over time, we can pre-compute them and store them offline. The results from the Item Based recommender is loaded to the Hadoop Distributed File System (HDFS) to have a scalable and fault-resistant storage. The User Based recommender results have to be computed every time during a recommendation since unlike items, ratings provided by users.

3. MODEL EVALUATION

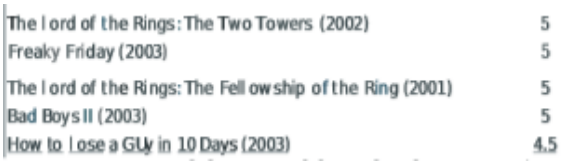
Qualitative evaluation: The movie recommender system built in this paper facilitates the understanding of how a recommender system works. To evaluate the accuracy and relevancy of the results produced by our system, we analyse both the approaches differently.

Movie 1	Movie 2	Similarity
1800421139	1800379216	0.99959636
1800061638	1800111258	0.99959064
1800121659	1800379216	0.99955463
1807537463	1804738128	0.9995 903
8028 3191	1807858489	0.9995346
1800121659	1800111258	0.9995051
1800061638	1800121659	0.999477 5
1800421139	1800121659	0.99944425
1800111258	1800379216	0.99939984
1800421139	1800111258	0.99938726
1800061638	1800379216	0.9993557
1800421139	1800061638	0.999335
1800080788	1800080795	0.9992829
180743259	1807428853	0.9992593

We compare the Item based similarity coefficient results as given in the above figure by mapping the Movie ID of Movie 1 and Movie 2 to their titles. As evident from the table, movies which are similar are given a higher similarity metric.

For user-based recommender system, we evaluate the model using the Average Absolute Difference Recommender Evaluator. We divide the training data into test and train samples. Next, we evaluate the rating predictions on test data against the actual ratings as specified in the training data. The figure below shows the raw output from the user-based filtering technique. The system recommends 10 movies to user and returns the nearest neighbours which have most similar taste preference as him. For each movie recommended, it also predicts the ratings by that user. We get an average absolute difference of 0 which proves that the predictions made on the ratings of the recommended items are 100% accurate.

The below figure shows the output that is obtained and kept before the user.



4. CONCLUSION:

In this paper we have implemented a movie recommendation system using collaborative filtering. It is implemented using Apache Mahout and takes the ratings given to movies to provide movie suggestions. Our system considers the user ratings to recommend movies. In the future, more features such as the genre of the movie, the directors, the actors and soon could be considered as well to provide suggestions. In addition, a new framework called Apache Prediction 10 could be looked into to develop the system instead of Mahout.

5. REFERENCES

[1] A. V. Dev and A. Mohan, "Recommendation system for big data applications based on set similarity of the user preferences" 2016 International Conference of on Next Generation Intelligent Systems(ICNGIS), Kottayam. 2016, pp.1-6. Doi: 10.1109/ICNGIS.2016.7854058

[2] Kumar, Manoj & Yadav, D.K. & Singh, Ankur & Kr, Vijay, "A Movie Recommender System: MOVREC", 2015 International Journal of Computer Applications. 124. 7-11. 10.5120/ijca2015904111

[3] S. G Walunj, K Sadafale, "An online recommendation system for e-commerce based on Apache Mahout framework", 2017 ACM SIGMIS International Conference on Computers and People Research, pp. 153-158,2013.

[4] 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), Ningbo, China, 2017, pp. 504-509. doi: 10.1109/ICMLC.2017.8108968

[5] Z. D Zhao, M. S Shang, "User-Based collaborative filtering recommendation algorithms on Hadoop", Proc. of Third International Workshop on Knowledge Discovery and Data Mining, pp. 478-481, 2016.

[6] B. Sarwar, G. Karypis, 1. Konstan, 1. Riedl, "Item-based collaborative filtering recommendation algorithms", Proceedings of the 10th international conference on World Wide Web, pp. 285-295, 2001

[7] Koen Verstrepen, Bart Goethals, "Unifying nearest neighbors collaborative filtering", Proceedings of the 8th ACM Conference on Recommender systems, October 06-10,2014, Foster City, Silicon Valley, California, USA doi:10.1145/2645710.2645731

[8] Jain, A., & Vishwakarma, S. K., "Collaborative Filtering for Movie Recommendation using Rapid Miner" International Journal of Computer Applications (0975 - 8887) Volume 169 - No.6, July 2017

[9] M. Gardener, "Statistics for Ecologists Using R and Excel (Edition 2) ". [Online]. Available: http://www.dataanalytics.org.uk [Accessed: 05-May-2018]

[10] Community Shared, "Apache Mahout". [Online]. Available: https://mahout.apache.org/ [Accessed:02-April-2018]