Recommender System using Collaborative Filtering and Demographic Characteristics of Users

Shano Solanki
Department of Computer Science
NITTTR,
Chandigarh (U.T.), India
s_solanki_2000@yahoo.com

Dr. Shalini Batra
Department of Computer Sc. and Engg.
Thapar University
Patiala, India
sbatra@thapar.edu

***Abstract*—**Recommender systems use variety of data mining techniques and algorithms to identify relevant preferences of items for users in a system out of available millions of choices. Recommender systems are classified into Collaborative filtering, Content-Based filtering, Knowledge-Based filtering and Hybrid filtering systems. The traditional recommender systems approaches are facing many challenges like data sparsity, cold start problem, scalability, synonymy, shilling attacks, gray sheep and black sheep problems. These problems consequently degrade the performance of recommender systems to a great extent. Among these cold start problem is one of the challenges which comes into scene when either a new user enters into a system or a new product arrives in catalogue. Both situations lead to difficulty in predicting user preferences due to non-availability of sufficient user rating history. The study proposes a new hybrid recommender system framework for solving new user cold-start problem by exploiting user demographic characteristics for finding similarity between new user and already existing users in the system. The efficiency of recommender systems can be improved by proposed approach which calculates recommendations for new user by predicting preferences within much smaller cluster rather than from the entire customer base. The analysis has been done using MovieLens dataset for enhancing the performance of online movie recommendation system.

Keywords- *Hybrid Recommender System, Demographic information, Collaborative filtering, Content-based filtering*.

*****

I. INTRODUCTION

Recommender systems are software applications or web portal that generates personalized preferences using information filtering techniques and algorithms with a goal to support in decision making by users in the system. Recommender systems are extensively used in various application domains i.e. online books ordering, online shopping, online hotel bookings, audio and video recommendations and so on.

People have been relying upon recommender systems for different reasons in their day to day life. The concept of recommender system is not new as it has been into practice for a long time though offline since there are so many professions and businesses, and the success of which is totally dependent upon their recommender system. The concept of online recommender system has gained popularity due to online availability of goods and services and the role of recommender system becomes more important when one kind of product or services are being offered by so many online service providers. Therefore recommender system has proved a boon for the society which helps them choose the best among the millions of available choices.

Recommender systems depends upon several factors like users ratings given to collection of items based upon their satisfaction level, their likes and dislikes, age, gender, occupation, region or locality, community etc. . Google, Amazon, Facebook, Twitter, YouTube, Flipkart, Pandora are some of the most popular websites that are using recommendation engine to filter choices based upon individual preferences. Use of a recommendation engine is becoming a standard element of a modern web presence[6].

From now onwards the paper is organised as Recommendation System Techniques, Recommendation System Challenges, Cold-Start problem solution using Demographic Characteristics: Proposed Methodology, Top-N Recommendation generation, Experiment results, Conclusion and future scope, Detailed diagram of Architecture of Proposed Methodology.

II. RECOMMENDATION SYSTEM TECHNIQUES

The recommender system uses several approaches for providing best results in the form of recommendations for an individual who is looking for preferences from an online recommender system. In the present times the following approaches are being used namely, Collaborative filtering (CF), Content-Based filtering (CBF), Knowledge-based filtering (KBF) and Hybrid Filtering (HF). [1]

A. COLLABORATIVE FILTERING

Collaborative-Filtering systems focus on the relationship between users and items. This method finds user-similarities using m x n rating matrix containing ratings data given by various users corresponding to same set of items in the online store. Here each entry in R

 ISSN: 2321-8169
4735 - 4741

user-item data matrix i.e. $R_{u,j}$ represents the rating given by user u to item j (as shown in Table 1).

TABLE 1: Showing User-item Matrix

	$Item_1$	$Item_2$	$Item_{....}$	$Item_i$	$Item_{....}$	$Item_n$
$User_1$	$R_{1,1}$	$R_{1,2}$	$R_{1,...}$	$R_{1,i}$	$R_{1,...}$	$R_{1,n}$
$User_2$	$R_{2,1}$	$R_{2,2}$	$R_{2,...}$	$R_{2,i}$	$R_{2,...}$	$R_{2,n}$
$User_{....}$	$R_{...,1}$	$R_{...,2}$	$R_{.....}$	$R_{...,j}$	$R_{.....}$	$R_{...,n}$
$User_u$	$R_{u,1}$	$R_{u,2}$	$R_{u,...}$	$R_{u,i}$	$R_{u,...}$	$R_{u,n}$
$User_{....}$	$R_{...,1}$	$R_{...,2}$	$R_{.....}$	$R_{...,j}$	$R_{.....}$	$R_{...,n}$
$User_m$	$R_{m,1}$	$R_{m,2}$	$R_{m,...}$	$R_{m,i}$	$R_{m,...}$	$R_{m,n}$

Collaborative filtering techniques are based upon either probabilistic models or non-probabilistic models. Nearest-neighbor algorithms are CF non-probabilistic algorithms[2]. It comes in two different forms i.e. User-based nearest neighbor and item-based nearest neighbor collaborative filtering algorithms. Further collaborative filtering is classified into two categories:

i) Memory-based collaborative filtering techniques use the entire or a sample of the user-item database to generate predictions. Every user belongs to a group of users with similar interests. The approach is based upon finding neighbours of an active user or new user in order to predict his/her preferences on a new item. Here, we use the nearest-neighbour based Collaborative filtering algorithms to predict the preferences or Top-N recommendations for the active user. Different kind of aggregate analysis can be done on similar users data in order to generate relevant recommendations, in certain priority order, if required.
Similarity can be computed item-based or user-based and there are several methods to calculate the similarity or distance or weight between users or items such as Minkowski distance, *Pearson correlation* and cosine-similarity metrics.

ii) Model-based collaborative filtering techniques allows the system to learn to recognize complex patterns based on the training data, and then make intelligent predictions for the collaborative filtering tasks for test data or real-world data, based on the learned models. Sometimes complete dataset is taken as training data and sometimes the dataset is spilt in a particular ratio to train the model and for testing purpose. Bayesian models, clustering models are examples of model-based CF. Usually; classification algorithms can be used as *CF* models if the user ratings are *categorical*, and regression models and *SVD* methods can be used for *numerical ratings* [2].

B. CONTENT-BASED FILTERING

This method finds the preferences of the current user about new item using rating history of current user related to previously used items. Similarity of items is determined by measuring the similarity in their properties. So, in this type of filtering method there is no dependency on rating records of other users in order to generate preferences for current user. Content-Based systems focus on properties of items. For example, if the user has purchased a book on amazon.com which uses recommender system then the user starts getting additional preferences for buying books from online book store which includes same or similar keywords information for books.

C. KNOWLEDGE-BASED FILTERING [1]

Knowledge based recommender systems makes use of knowledge structure to make inference about the user needs and preferences. Such kind of recommender systems have knowledge about what kind of items are liked by a user based upon user profile information and context-related information. So, a relationship can be established between user needs and relevant recommendation out of available collection of items for that user.

D. HYBRID FILTERING

No single recommender system approach is found to be efficient enough to generate relevant and accurate recommendation preferences, so, a hybrid recommender systems came into existence to overcome the limitations of traditional recommendation approaches mentioned above. These systems are based upon combining advantages of more than one traditional approach for recommendation generation for example; collaborative filtering approach with content-based approach or collaborative filtering with demographic characteristics based recommendation approach etc.[1].

III. RECOMEMNDER SYSTEM CHALLENGES

Recommender systems have been used extensively for generating recommendations using various recommendation techniques in multifarious application domains and this has brought into notice many

challenges. Research areas are emphasizing on solving the issues mentioned below:

- **Data Sparsity** problem arises if the user-item matrix containing ratings details is extremely sparse and this situation further leads to inefficient recommender systems which are based upon **nearest-neighbor algorithms** for calculating user similarity. This problem is further classified as **reduced coverage problem and neighbor transitivity problem[2].**
- **Scalability** problem refers to a situation when numbers of items and users giving ratings those items increased tremendously and it becomes difficult for a recommender system to handle such a big data due to computational complexity and constrained resources. So, it goes beyond the limit of acceptability of recommender systems.
- **Synonymy** means recommender system fails to recognize the similarity among two items when some similar items have different names and the recommender system treats them as if they are different items. This leads to problem of recommending similar items, called as synonymy problem[1].
- **Gray sheep** problem arises because the user's choice does not match with any other user or group of users in agreement or disagreement consistently.
- **Black sheep** problem refers to a situation when the user's choice correlates with very few users or no users at all. In such cases recommender system proves to be inefficient or fruitless in generating preferences[1].
- **Shilling attacks** are categorized into **push attacks and nuke attacks.** When the competitor vender makes use of unfair ways and means to show more rating of their own items as compared to other vendor products then it is known as push attacks. On the other hand if they try to reduce the rating of their rivals or competitors then it is called as nuke attacks.
- **Cold-start problem** means when a recommender system is unable to generate or predict ratings due to initial lack of ratings then it is referred to as cold-start problem. This kind of situation happens when either a new user arrives into a system having no rating records available with recommender system or when a new item enters into system and till now no one has given rating to that item. So, it becomes difficult for a recommender system to generate choices for a new user and hence the objective of recommender systems is not achieved[4].

IV. COLD-START PROBLEM SOLUTION USING DEMOGRAPHIC CHARACTERISTICS: PROPOSED METHODOLOGY

Several efforts has been done till now for solving this problem of recommender systems using several approaches. One approach is to make use of optimistic exponential type of ordered weighted averaging (OWA) operator to fuse the results of more than one approaches or recommendation strategies to gain performance over single approach[5]. Many researchers are using user behavior study to recommend items according to user choices. Moreover, social tagging system on social networking sites is quite popular now a days for generating recommendations based upon similarity index with friends and relatives from social media[7]. Constraint-based, Locality or Context-aware information based recommendation systems are also key areas of research[10].

In the proposed methodology the main focus is on solving user cold-start using demographic information based user-similarity[3]. The demographic information is composed of various user characteristics like Age, Gender, Occupation, Zip-Code, race, religion, marital-status, locality, hobbies etc. This approach is preferred when user rating history is not available for generating preferences. So, this work is effective for finding recommendations using external information provided by users at registration time.

The main data mining techniques used here are K-means clustering algorithm for finding user similarity based upon users demographic characteristics like UserID, Age, Gender and Occupation and then evaluating clusters performance using J48 classification algorithm. AddCluster filtering algorithm is used for assigning clusters to instances in user dataset because of unavailability of class labels for instances. It is a form of unsupervised machine learning. The proposed approach is divided into two parts:

i) OFFLINE ANALYSIS USING WEKA TOOL

- First of all, dataset specific to problem domain is refined for testing purpose.
- Clusters are assigned to instances in the dataset using AddCluster fitering algorithm.
- Finally, clusters are evaluated using J48 classification algorithm.
- Above process helps in finding which combination of demographic attributes can be used for accurate classification of instances and for generating decision rules for identifying cluster for new user in cold –start situation. This research has depicted UserId, Gender

ISSN: 2321-8169
4735 - 4741

and Age as more appropriate combination for finding user-based similarity due to more no. of correctly classified instances using this combination of attributes as shown in Table 3.

ii) ONLINE RECOMMENDATION GENERATION

- Here, neighborhood consisting of similar users to new user who has just arrived in the system is found using k-means algorithm.
- Top-N recommendations are predicted on the basis of users in neighborhood using MySql database integration.
- Finally, recommendations are generated for the new user online.

V. TOP-N RECOMMENDATION GENERATION

It means to recommend a set of N top-ranked items that will be of interest to a particular user. It is based upon analysing the user-item rating matrix to find relation between different users or items for computing the recommendations. Top-N recommendations are further generated using either user-based or item-based Top-N recommendations[2].

In proposed approach recommendations are generated by using MySql database created for MovieLens dataset and then respective queries are run to find required recommendation for new user[9].

The sample dataset is consisted of 500 records from users.csv file and saved as user2.csv file. Another refinement is done on movie.csv dataset consisted of 1200 movies[11]. The ratings.csv file contains user rating records corresponding to movie ratings given by 500 users on 1200 movies and rest of the records are not considered. The tables under experimentation are imported using PhpMyAdmin and under a new Mysql database with the name **dbmovielens**.

QUERY1: To find out the user rating based movie recommendations.

QUERY2: To find out the user rating based movie recommendations after filtering the data on the basis of cluster information for the new user in the system whose rating record is not available.

TABLE 2: Description of features Age, Gender and Occupation in MovieLens Dataset [8]

Gender	
0	Male
1	Female

Age	
0	<18
18	18-24
25	25-34
35	35-44
45	45-49
50	50-55
56	56+

Job			
0	other or not specified	11	lawyer
1	academic / educator	12	programmer
2	artist	13	retired
3	clerical/admin	14	sales/marketing
4	college/grad student	15	scientist
5	customer service	16	self-employed
6	doctor/health care	17	technician/engineer
7	executive/managerial	18	tradesman/craftsman
8	farmer	19	unemployed
9	homemaker	20	writer
10	K-12 student		

VI. EXPERIMENT RESULTS

Experiments are performed on MovieLens Database offline to find how classifications of instance based upon user demographic characteristics can be more accurate. For this purpose J48 classification algorithm is used and its result is shown for three different experiments based upon different combination of attributes.

TABLE 3: Comparison statistics of clustering on users.csv dataset

Experiment details	Classification details
J48 classifier Using 4 attributes (UserId, Age, Gender, Occupation)	Correctly Classified Instances 6030 99.8344 % Incorrectly Classified Instances 10 0.1656 %
J48 classifier Using 3 attributes (UserId, Age, Gender)	Correctly Classified Instances 6037 99.9503 % Incorrectly Classified Instances 3 0.0497 %
J48 classifier Using 3 attributes (UserId, Age, Occupation)	Correctly Classified Instances 6030 99.8344 % Incorrectly Classified Instances 10 0.1656 %

TABLE 4: Comparison between movie recommendations based on Query results 1 and 2

Movieid from first query (on user rating info based)	Movieid from second query (user demographic cluster specific)
318	318
296	527
527	296
593	50
110	593
50	110
260	260
858	608
1198	858
608	356

Let recommendations are provided gender specific i.e. for male or female for the new user with specific details given below:

UserId	Age	Gender	Occupation
502	F	30	4

The new user belongs to say cluster 3| as per k-means clustering algorithm and then recommendations are generated from neighborhood consisted of cluster 3 other users

Table 5: Top10 recommendations for new female visitor

Movieid	rating	female_count
318	5.0	10 [->]
110	5.0	9 [->]
356	5.0	8 [->]
527	5.0	8 [->]
593	5.0	8 [->]
47	5.0	7 [->]
296	5.0	7 [->]
1	5.0	7 [->]
590	5.0	7 [->]
858	5.0	7 [->]

Similarly, movie recommendations for male can be found using his similarity with users in specific cluster to which he/she belongs to. Here recommendation are presented based upon highest frequency count of MovieId with rating 5 as movies are rated between 1-5 numeric rating and 5 is considered as highest preference and 1 as lowest ranking.

VII. CONCLUSION AND FUTURE SCOPE

The proposed approach has resulted in reduction of overall execution time for generating recommendations and deals well with user cold-start problem. If n is the no. of users and m is the no. of items then the user-item rating matrix will be of the order of n*m and if we need to compute recommendations on user rating based recommendations then its complexity will be of the order of O(mn). The no. of ratings in total is quite high as compared to no. of users[4]. The proposed methodology first of all analyzes the right combination of attributes for grouping dataset instances into clusters and then provides a sample of similar users on the basis of information contained in demographic attributes. These records constitutes neighborhood and as a consequence the search space for recommendation generation has reduced to great extent.

The aggregate function used for calculating Top-N recommendations is frequency count of users who rated a particular movie with highest score i.e. 5. The quality of recommendation generation also depends upon rating distribution and type of aggregate analysis. Rather than taking simple count based on highest rating frequency count another kind of aggregate functions can also be used say for example average rating for a particular movie.

Moreover, another set of demographic attributes can be exploited for finding clusters and hence recommendation accuracy can be improved. For this demographic attribute collection in user profiles can be increased for getting better recommendations. The information about user likes/dislikes, cultural background and other attributes can be used for personality diagnosis and community detection which is quite popular among researchers now a days for recommendation generation for an individual or for a group with same preferences.

In future, this methodology can be applied to other problem domains for generating recommendations and can also be used for new item cold-start problem. The dataset can be extended for analyzing the performance of recommender system in real-time. The only limitation in this approach is dependency upon user profile data and many users feels irritated and impatient in filling long forms. This limitation may further be solved using social tagging system in social networks.

ISSN: 2321-8169
4735 - 4741

VIII. DETAILED DIAGRAM OF ARCHITECTURE OF PROPOSED METHODOLOGY

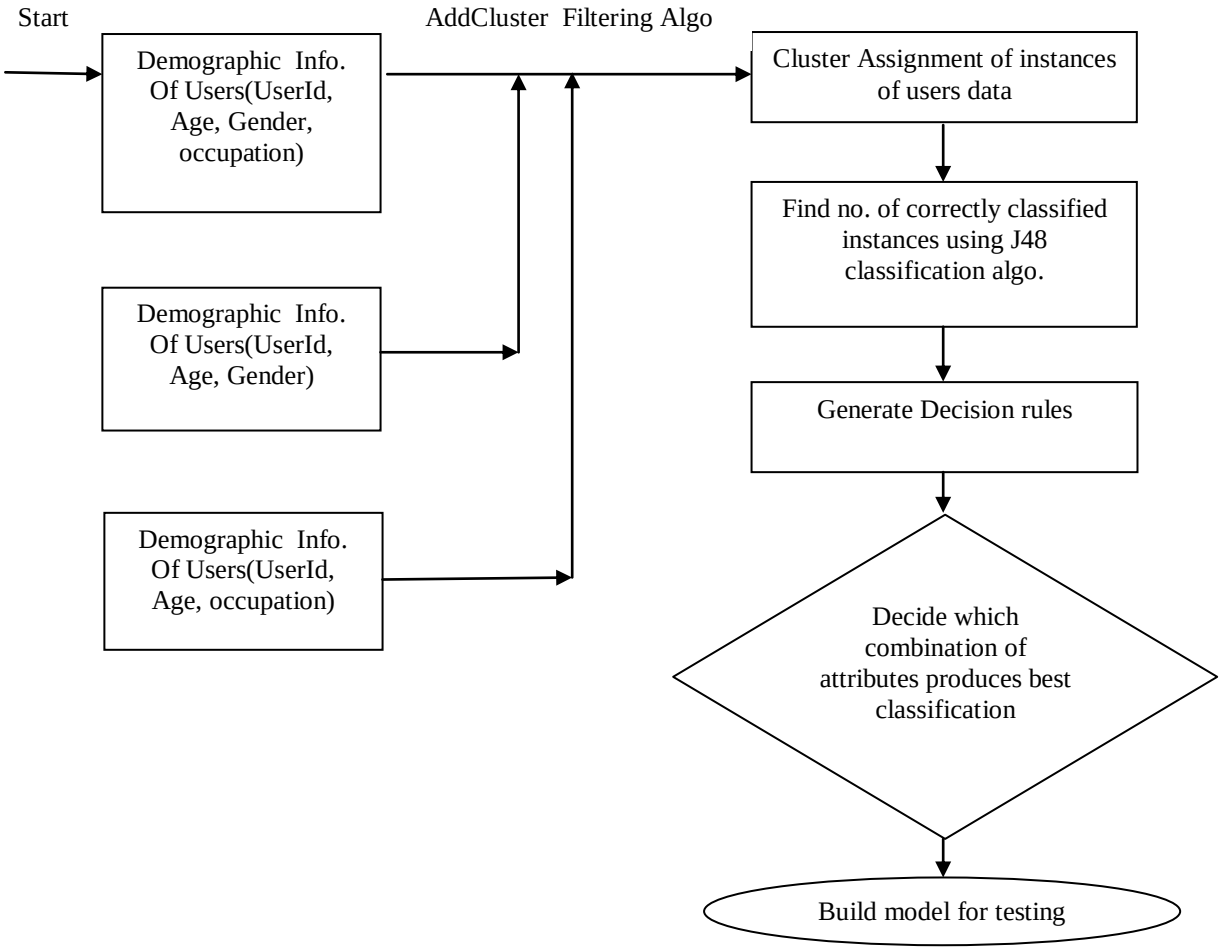

Figure 1: Offline Analysis using Weka Tool

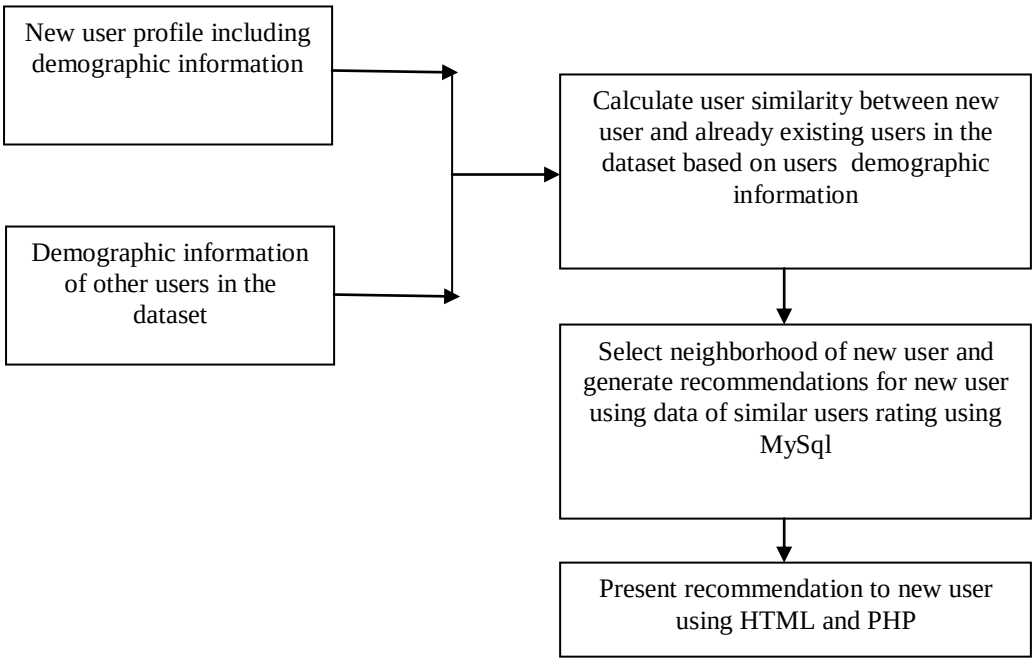

Figure 2: Online Recommendation Generation

REFERENCES

[1] Tranos Zuva, Sunday O. Ojo, Seleman M. Ngwira, and Keneilwe Zuva, "A Survey of Recommender Systems Techniques, Challenges and Evaluation Metrics," *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 11, November 2012.

[2] Xiaoyuan Su and Taghi M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," *Advances in Artificial Intelligence*, vol. 2009.

[3] Laila Safoury and Salah Akram. (2013, August) Exploiting User Demographic Attributes for Solving.

[4] Honey Jindal and Sandeep Kumar Singh, "A HYBRID RECOMMENDATION SYSTEM FOR COLD-START PROBLEM USING ONLINE COMMERCIAL DATASET," *International Journal of Computer Engineering and Applications*, vol. VII, no. 1, July 2014.

[5] Javad Basiri, Azadeh Shakery, Behzad Moshiri, and Zi Morteza Hayat, "Alleviating the Cold-Start Problem of Recommender Systems Using a New Hybrid Approach," *5th International Symposium on Telecommunications (IST'2010)*, 2010.

[6] Michael Hahsler, "recommenderlab: A Framework for Developing and Testing Recommendation Algorithms".

[7] Zi-Ke Zhang, Chuang Liu, Ye-Cheng Zhang, and Tao Zhou, "Solving the cold-start problem in recommender systems with social tags," *EPL*, October 2010.

[8] Xuan Nhat Lam , Thuc Vu , Trong Duc Le , and Anh Duc Duong , "Addressing Cold-Start Problem in Recommendation," in *2nd International Conference on Ubiquitous Information Management and Communication, ICUIMC 2008, Suwon, Korea, January 31 - February 01*, 2008.

[9] Daniel Lemire and Sean McGrath, "Implementing a Rating-Based Item-to-Item. Recommender System in Php/Sql," 2013.

[10] Amit Sharma. What are some of the most interesting research problems in Recommender Systems? [Online]. HYPERLINK "http://www.quora.com/What-are-some-of-the-most-interesting-research-problems-in-Recommender-Systems" http://www.quora.com/What-are-some-of-the-most-interesting-research-problems-in-Recommender-Systems

[11] www.grouplens.org. [Online]. "http://grouplens.org/datasets/movielens/." http://grouplens.org/datasets/movielens/.