International Journal on Emerging Technologies (Special Issue NCETST-2017) **8**(1): 55-61(2017)
(Published by Research Trend, Website: www.researchtrend.net)

ISSN No. (Print) : 0975-8364
ISSN No. (Online) : 2249-3255

Solving Cold-Start Problem in Recommender System Using User Demographic Attributes

Gaurav Agarwal[1], Dr. Himanshu Bahuguna[2] and Dr. Ajay Agarwal[3]
[1]Research Scholar, Uttarakhand Technical University, Dehradun, (U.K.), INDIA
[2]Professor, Shivalik College of Engineering, Dehradun, (U.K.), INDIA
[3]Professor, Krishna Institute of Engineering & Technology, Ghaziabad, (U.P.), INDIA

ABSTRACT: Recommender systems have been used tremendously academically and commercially, recommendations generated by these systems aim to offer relevant interesting items to users. Several approaches have been suggested for providing users with recommendations using their rating history, most of these approaches suffer from new user problem (cold-start) which is the initial lack of items ratings. This paper suggest utilizing new user demographic data to provide recommendations instead of using rating history to avoid cold-start problem. We present a framework for evaluating the usage of different demographic attributes, such as age, gender, and occupation, for recommendation generation. Experiments are executed using MovieLens dataset to evaluate the performance of the proposed framework.

Index Terms - Demographic filtering, information retrieval, personalization, recommender system.

I. INTRODUCTION

In the recent years, recommender system has been used tremendously academically and commercially providing users with items (i.e.: products, services, or information) which match their preferences and interests. These items are recommended by the system to guide user in a personalized way based on user's historical preferences to discover unseen items among a great collection of items stored on the system. Recommender systems are utilized in different domains to personalize its applications by recommending items, such as books, movies, songs, restaurants, news articles, jokes, among others.

Researchers have suggested several approaches for building recommender systems which offer items differently to users based on a specific assumption in order to match their interests. Nevertheless, all recommendation approaches have strengths and weaknesses that should be considered while choosing the most suitable approach to implement. Therefore, hybrid recommenders are commonly used for combining two or more recommendation approaches together earning better performance and fewer drawbacks [1].

The recommendation system types can be distinguished into two most commonly used recommendation approaches:

(A). Content-based filtering method: Content-based filtering methods are based on a description of the item and a profile of the user's preference. These algorithms try to recommend items that are similar to those that a user liked in the past (or is examining in the present). In particular, various candidate items are compared with items previously rated by the user and the best-matching items are recommended. like as shown in the fig. 1.

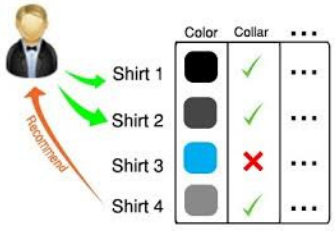

Fig. 1. Content-based Filtering.

The fig. 1 shows that there are four shirts with different features (like color, collar, ...). Here the user liked to the Shirt 1 and Shirt 2 (Both Shirt 1 and Shirt 2 have collar features). As per Fig. 1, Shirt 4 has also the Collar feature, So, the content based approach recommend the Shirt 4 to the user which is similar to the Shirt 1 and Shirt 2(as all three shirts-1,2 and 4 have the collar feature).

(B). Collaborative filtering method: Collaborative recommender provides recommendations based on users' similarity, it assumes that users with similar tastes will rate items similarly [2, 3]. It attempts to find users having similar rating history to the target user (user who requires recommendations), building a

neighborhood from which the recommended items are generated.

Fig. 2. Collaborative filtering.

Fig. 2 shows that there are two persons both liked the similar items (like pizza and namkeen dish). Here the 2nd person (who is in stand-up form) also liked the Cold drink. So, collaborative filtering methods recommend the cold drink to the other user (who is on bicycle)

However, these approaches had been addressed to suffer from new user problem, known as cold start problem, which is having initial lack of ratings when a new user join the system [4]. Since both approaches assumption are based upon user's ratings history, this problem can significantly affect negatively the recommender performance due to the inability of the system to produce meaningful recommendations [5,6]. Hence, an alternative kind of input is required to be obtained explicitly from users to be utilized for suggesting recommendations instead of ratings.

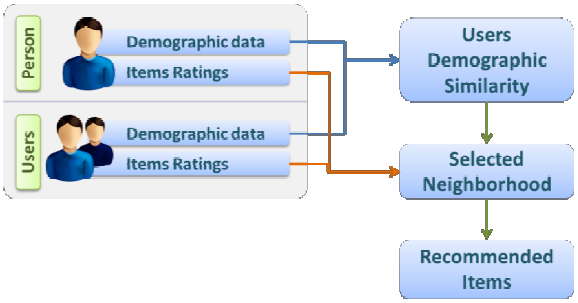

Fig. 3. Demographic-based approach.

Another recommender approach had been introduced which utilizes user demographic data as an alternative input for recommender system which is known as demographic-based approach.

Demographic-based recommender, as shown in Fig. 3, suggests utilizing users' demographic data stored on their profiles (i.e. age, gender, location … etc.), it assumes that users with similar demographic attribute(s) will rate items similarly [8]. This recommender obtains group of user having similar demographic attribute(s) forming a neighborhood from which newly recommended items are generated. In this paper we provide a framework for evaluating users' demographic attributes to be used in generating recommendations for new users.

The rest of the paper is structured as follows. Section II shows other researchers work applying demographic approach in recommender system. In Section III, the framework developed in this paper is described. Section IV explains the experiment conducted to evaluate the proposed framework.

Finally, Section V concludes the paper and provides directions for future research [11].

II. RELATED WORK USING DEMOGRAPHIC RECOMMENDER SYSTEM

The demographic-based and collaborative filtering approaches hybridization had been introduced by researches for improving the recommendation quality rather than solving "cold-start problem". A group of researchers have applied a hybrid model-based approach on movie domain using user demographic data to enhance the recommendation suggestion process, it classified the genres of movies based on user demographic attributes, such as user age (kid, teenager or adult), student (yes or no), have children (yes or no) and gender (female or male) [12,13].

Additionally, other researchers modified user similarity calculation method to employ the hybridization of demographic and collaborative approaches. A modification to k-nearest neighborhood had been introduced which calculates the similarity scores between the target user and other users forming a neighborhood, increasing the scores of users having similar ratings and demographic attribute (each demographic attribute had been evaluated along similar ratings separately) [15]. Whereas another research work demonstrated another modified version of k-nearest neighborhood by adding a user demographic vector to the user profile, the similarity calculation consider both ratings and demographic vector (holding all of the demographic attributes) .

In contrast, this paper suggest a novel framework to resolve the new user "cold-start" problem by utilizing the demographic data explicitly given by a user. The framework aims at evaluating the influence of demographic attributes on the user ratings, to assist the recommender system designer to improve recommendations quality for new users. The framework had been examined using a movie dataset to evaluate the generated recommendations accuracy and precision.

III. DEMOGRAPHIC ATTRIBUTES EVALUATION FRAMEWORK FOR RECOMMENDER SYSTEM

The demographic-based recommendation process performs three stages: data input, similarity calculation and recommendation calculation (as shown in Fig. 3). Data input is the stage which holds new target user's demographic data (the user who requires recommendations) and also ratings and demographic data of the rest of users. Similarity calculation stage utilizes users' demographic data to obtain a number of users having similar demographic data to the target user forming a neighborhood. Finally, Recommendation calculation stage obtains items which have been commonly positive-rated by neighborhood users to be suggested to the target user [17,19].

Furthermore, the similarity calculation stage requires selecting the demographic attributes to be used for calculating the similarities. For instance, Table 1 demonstrates the demographic data of four users; each user has four demographic attributes (gender, occupation, country and age). Let us assume that John is a new user who demand recommendation, the system has to calculate the similarity between John and other users based on the selected attributes [22].

Table 1: Example of Users Demographic Data.

Name	Gender	Occupation	Country	Age
Raj	M	Student	France	13
Mohan	M	Doctor	France	34
Sachin	F	Student	USA	12
Deepak	M	Teacher	France	27

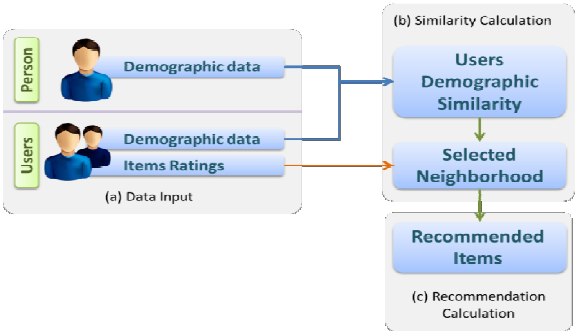

Fig. 4. Demographic-based approach for new users.

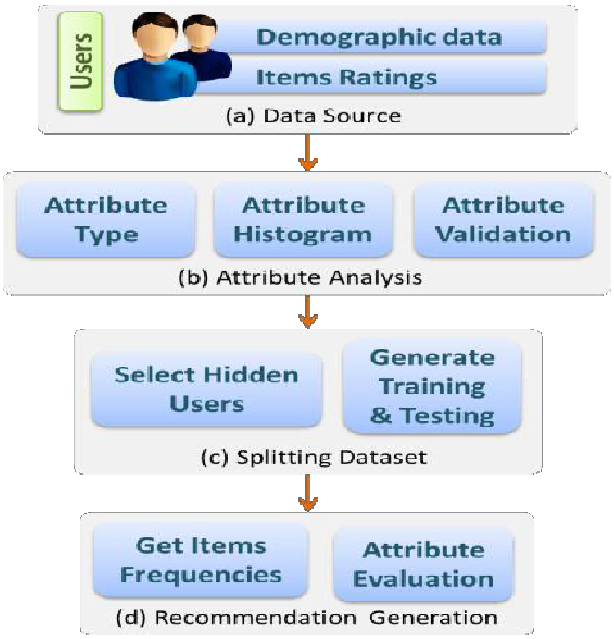

Fig. 5. Demographic attributes evaluation framework.

The similarity calculation output depends on the way the system interprets how users are similar, if users having the same occupation are similar then Sarah is similar to John, else if users having the same gender and nationality are similar then Paul and Mike are similar to John. Therefore, the choice of attributes affects the similarity calculation output which consequently influences the results of recommendation calculation stage. The proposed framework consists of four modules: data source, attribute analysis, splitting dataset and recommendation generation (as shown in Fig. 5). Data source has all data about users (demographic and items ratings) stored. Attribute analysis module works on analyzing the type of demographic attributes, the distribution of attributes values across the dataset (histogram) and validity of using these attributes for recommendations [23]. Splitting dataset module splits training and testing for each valid attribute by removing all the ratings of a few randomly selected users (considered as hidden/new users who have no ratings) from training file and adds their ratings to a testing file. Afterwards, Recommendation generation module extracts most frequent items appeared in the training file (rated by users having similar attribute value to the hidden users) recommending them to the new users (hidden users), the testing file is used for evaluating the correctness of the recommendations compared to the hidden ratings.

IV. EXPERIMENTAL METHODOLOGY

The framework had been experimented using the publicly available data of GroupLens movie recommender system, MovieLens data set (http://www.grouplens.org/node/73). This dataset had been used by many researchers; some researchers used the dataset to execute their experience. While others used the MovieLens dataset to study the state-of-art of recommender systems applying collaborative approach different techniques [25]. Additionally, GroupLens provides various versions of the dataset, such as: MovieLens 100k, MovieLens 1M, and MovieLens 10M datasets.

A. Data Source

The dataset used in this paper is MovieLens 100k; it consists of 100,000 ratings which were evaluated by 943 users on 1682 movies. Each user had rated at least 20 movies; the ratings are assigned numerically from 1(bad) to 5(excellent). Table 2 shows information about MovieLens dataset files used in the experiment.

Table 2: Movielens Dataset Information.

MovieLens Dataset Files	File Attributes Description
u.user	The user file contains demographic information about the 943 users. "user id | age | gender | occupation | zip code "
u.item	The item file holds information about the items(movies). "movie id | movie title | release data | video release data | IMDb URL | Unknown | Action | Adventure | Animation | Children's | Comedy | Crime | Documentary | Drama | Fantasy | Film-Noir | Horror | Musical | Mystery | Romance | Thriller | War | Western "
u.data	The data file contains 100,000 ratings by 943 users on 1682 items. "user id | item id | gender | rating | timestamp "

B. Attribute Analysis

The attribute analysis module determines the type and value ranges of the demographic attributes in Movielens dataset, shown in Table 3. Then, the frequency of each attribute value is calculated showing the number of users having similar value; Fig. 4 illustrates the histogram of Movielens demographic attributes except for zip code which had only 148 low frequent duplicated values. Afterwards, the module validates the attributes that can be used for recommendations by checking the following conditions:

1) **Invalid Value Range:** It occurs when some ranges of the demographic attribute has low frequency, such as the frequency of age attribute "less of equal nine years old" range has only one user within its range (Fig. 4 (b)), therefore, the system will not be able to offer recommendations for new users who fall in this range using age attribute.
2) **Invalid Attribute Value:** It exists when a value of an attribute has a vague meaning, such as occupation attribute values "none" and "other" (Fig. 4 (c)), if more than one user had their occupation filled as "none" or "other" it doesn't imply that they are having similar taste or will rate items similarly.
3) **Invalid Attribute:** Attribute is considered invalid when its values are highly sparse, such as zip code attribute most of its values are distinct while a few has low frequency.

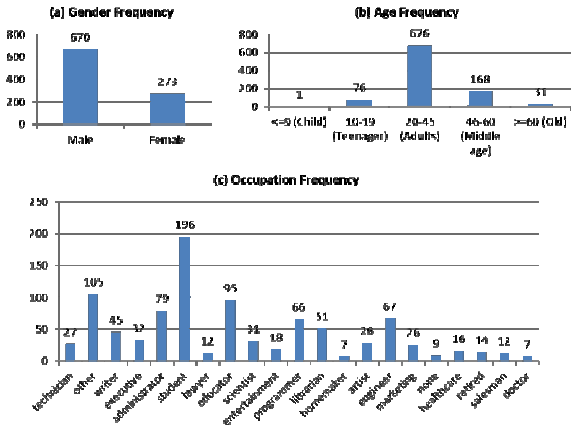

Fig. 6. Attributes Demographic.

C. *Splitting Dataset*

Table 3: Attribute Types.

Attribute Name	Data Type	Value Ranges
Gender	Character	M,F
Age	Number	7-73
Occupation	Text	21 Occupations
Zip Code	Text	695 distinct value

Table 4. Number of Hidden Users. Per

Attribute	Testing Users Number	Total No.
Age	Teenager (10-19) = 10	40 users
	Adults (20-45) = 20	
	Middle age (46-60) = 15	
	Old (>60) = 5	
Gender	20 per 2 genders	40 users
Occupation	10 per 4 occupations	40 users

Splitting dataset module creates training and testing files for each of the three valid attributes (age, gender and occupation) and their valid values (excluding "age" invalid range and "occupation" two invalid values) to be evaluated. Training dataset requires selecting number of users to hide their ratings adding them to testing dataset. Table 4 illustrates the number of users whom their ratings will be removed from training dataset (40 users per attribute); in our experiment only the most four frequent values of occupation attribute: Student, Educator, Administrator, and Engineer, will be considered while the rest of occupations will be excluded for sake of decreasing the number of trials [28, 30].

D. *Recommendation Generation*

The Recommendation generation module utilizes the training file of each attribute to calculate the frequency of all items rated by users having similar attribute value. For instance, the module uses gender training file to calculate the frequency of items rated by female gender and vice versa for male gender.

V. CONCLUSION AND FUTURE WORK

In this work we have presented a novel framework for evaluating demographic attributes available in recommender systems datasets to be used for recommending relevant items to new users. The framework was examined using MovieLens dataset, the experimental results of the dataset showed that all attributes have almost the same influence. Conclusively, it seems that the demographic data in the MovieLens dataset does not influence differently on users' ratings.

Further research can be performed to enhance the results, such as creating more than one training and testing dataset to be evaluated and gather the average of the results. Also a higher level of movie recommendation can be obtained by relating the movie genres to demographic attributes. Finally, this framework can be applied on different domains datasets.

REFERENCES

[1]. Pan Zhou, Yingxue Zhou, Dapeng Wu, Hai Jin, *Seni,* "Differentially Private Online Learning for Cloud-Based Video Recommendation With Multimedia Big Data in Social Networks", IEEE TRANSACTIONS ON MULTIMEDIA, VOL. **18**, NO. 6, JUNE 2016, pp. 1217-1219.

[2]. Yuan Cheng, Jaehong Park, and Ravi Sandhu, "An Access Control Model for Online Social Networks Using User-to-User Relationships", IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL. **13**, NO. 4, JULY/AUGUST 2016, pp. 424-425

[3]. Hanhua Chen, Hai Jin, and Shaoliang Wu, "Minimizing Inter-Server Communications by Exploiting Self-Similarity in Online Social Networks", IEEE TRANSACTIONS PARALLEL AND DISTRIBUTED SYSTEMS, VOL. **27**, NO. 4, APRIL 2016, pp. 1116 -1125.

[4]. Xiao-Lin Zheng, Senior, Chao-Chao Chen, Jui-Long Hung,

Wu He, Fu-Xing Hong, and Zhen Lin, "A Hybrid Trust-Based Recommender System for Online Communities of Practice", IEEE TRANSACTIONS ON LEARNING TECHNOLOGIES, VOL. **8**, NO. 4, OCTOBER-DECEMBER 2015.

[5]. Valeryia Shchutskaya, "Big Data Behind Recommender Systems", March 29, 2016 [online] available at: https://indatalabs.com/blog/data-science/big-data-behind- recommender-systems.

[6]. Linke Guo, Chi Zhang, and Yuguang Fang, "A Trust-Based Privacy-Preserving Friend Recommendation Scheme for Online Social Networks", IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL. **12**, NO. 4, JULY/AUGUST 2015, pp. 413-415.

[7]. M. Jones, "Introduction to approaches and algorithms for Recommendation systems" December 12, 2013 [online] available at: https://www.ibm.com/developerworks/library/os-recommender.

[8]. Hanhua Chen, Member, IEEE, and Hai Jin, Fan Zhang,"CBL: Exploiting Community based Locality for Efficient Content Search Service in Online Social Networks", IEEE TRANSACTIONS ON SERVICES COMPUTING, 2015, pp. 1 -12.

[9]. Esther Palomar, Lorena González-Manzano, Almudena Alcaide, Álvaro Galán, "Implementing a privacy-enhanced attribute based credential system for online social networks with co-ownership management", The Institution of Engineering and Technology 2016, pp. – 60- 68.

[10]. Yifeng Zeng, Xuefeng Chen, Yew-Soon Ong, Jing Tang and Yanping Xiang," Structured Memetic Automation for Online Human-like Social Behavior Learning", IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION [Online]. Available: http://dx.: doi.org/ 10.1109/TEVC.2016.2577593.

[11]. Wikipedia, "Recommender System" [online] available at: https://en.wikipedia.org/wiki/Recommender_system Hybrid_Recommender_Systems.

[12]. Ruchika, Singh, Ajay Vikram, Sharma Dolly, "Evaluation Criteria for Measuring the Performance of Recommender Systems", IEEE 2015.

[13]. Amazon.com [Online]. Available: http://en.wikipedia.org/wiki/Amazon.com

[14]. R. Chen, E. K. Lua, and Z. Cai, "Bring order to online social networks," in Proc. IEEE INFOCOM, 2011, pp. 541–545.

[15]. M. Srivatsa, L. Xiong, and L. Liu, "TrustGuard: Countering vulnerabilities in reputation management for decentralized overlay networks," in Proc. 14th Int. Conf. WorldWideWeb, 2005, pp. 422–431.

[16]. D. N. Kalofonos, Z. Antonious, F. D. Reynolds, M. Van-Kleek, J. Strauss, and P. Wisner, "MyNet: A platform for secure p2p personal and social networking services," in Proc. IEEE 6th Annu. Int. Conf. Pervasive Comput. Commun., 2008, pp. 135–146.

[17]. S. Wen, W. Zhou, J. Zhang, Y. Xiang, W. Zhou, and W. Jia , "Modeling propagation dynamics of social network worms," IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 8, pp. 1633–1643, Aug. 2013.

[18]. A. Sridharan, Y. Gao, K. Wu, and J. Nastos, "Statistical behavior of embeddedness and communities of overlapping cliques in online social networks," in Proc. INFOCOM, 2011, pp. 546–550.

[19]. Laila Safoury and Akram Salah, "Exploiting User Demographic Attributes for Solving Cold-Start Problem in Recommender System", Lecture Notes on Software Engineering, Vol. 1, No. 3, August 2013.

[20]. Mili Mohan, Robert S, "Alleviating Cold-Start Problem in LARS* Using Hybrid Systems", in IJIRCCE, Vol. 3, Issue 7, July 2015.

[21]. Mohammad Daoud, S.K Naqvi, Tahir Siddqi, "An Item-Oriented Algorithm on Cold-start Problem in Recommendation System", International Journal of Computer Applications (0975 – 8887), Volume 116 – No. 11, April 2015.

[22]. Melville, P. and Sindhwan, V. (2010) Recommender Systems. IBM T.J. Watson Research Centre, Yorktown Heights, NY. http://vikas.sindhwani.org/recommender.pdf.

[23]. Huang, Z., Chen, H. and Zeng, D. (2004) Applying Associative Retrieval Techniques to Alleviate the Sparsity Problemin Collaborating Filtering. *Transaction on Information Systems*, No. 1, 116-142.

[24]. Park and Tuzhilin 2008 Park, Y.-J. and Tuzhilin, A. 2008. The long tail of recommender systems and how to leverage it. In Proc. of the 2008 ACM Conf. on recommender systems. 11-18.

[25]. Pan Zhou, Member, IEEE, Yingxue Zhou, Student Member, IEEE, Dapeng Wu, Fellow, IEEE, and Hai Jin, Senior Member, IEEE, "Differentially Private Online Learning for Cloud-Based Video Recommendation With Multimedia Big Data in Social Networks", IEEE TRANSACTIONS ON MULTIMEDIA, VOL. **18**, NO. 6, JUNE 2016.

[26]. F. McSherry and K. Talwar, "Mechanism design via differential privacy", in *Proc. 48th Annu. IEEE Symp. Found. Comput. Sci.*, Oct. 2007, pp. 94–103.

[27]. Schafer, J.B., Konstan, J.A. and Riedl, J. (2001) E-Commerce Recommendation Applications. *Data Mining and Knowledge Discovery*, **5**, 115-153. http://dx.doi.org/10.1023/A:1009804230409.

[28]. A. Samuel, M. I. Sarfraz, and H. Haseeb, "A framework for composition and enforcement of privacy-aware and context-driven authorization mechanism for multimedia big data," *IEEE Trans. Multimedia*, vol. **17**, no. 9, pp. 1484–1494, Sep. 2015.

[29]. R. Fogues, J. M. Such, A. Espinosa, and A. Garcia-Fornes, "Open challenges in relationship-based privacy mechanisms for social network services," International Journal of Human-Computer Interaction, vol. **31**, no. 5, pp. 350–370, 2015.

[30]. S. Jiang, X. Qian, and J. Shen, "Author topicmodel-based collaborative filtering for personalized POI recommendations," *IEEE Trans. Multimedia*, vol. 17, no. 6, pp. 907– 918, Jun. 2015.