Trail Traveller

[1]Kavita Jain, [2]Kshitij Karekar,[3]Janeya Limbachia, [4]Nadar Jaysalviya Salwin

[1]Project guide, [2]Student, [3]Student, [4]Student
[1]Computer Engineering,
[1]Xavier Institute of Engineering, Mumbai, India.

Abstract : **A user based travel guide recommendation system is proposed in this paper. We have proposed a novel method to generate a hybrid recommendation list for the user. Technologies mainly used for recommendation systems are collaborative filtering. Collaborative filtering are of two types: User - User and Item - Item. Item based filtering technique is preferred in recommender systems; however it gives a monotonous list. We have proposed a solution where in we will combine both user - user and item - item filtering and get a hybrid recommendation. The unique part of this recommender system will be that it will use click stream for foreign and PCA for internal user behavior. The proposed system will be efficient and predict better than the traditional recommender systems.**

***Index Terms* – Cosine Similarity, K-Nearest Neighbor(KNN), Collaborative Filtering(CF), Item-Based Collaborative Filtering(IBCF), User-Based Collaborative Filtering(UBCF), PCA, Click-Stream, Machine Learning(ML), Android Application.**

I. INTRODUCTION

Lately, the demands for recommendation services have severely increased due to the massive flow of new content on to the internet. In order for users to find the content they desire, competent recommendation services are extremely helpful. Using ML and adding better methods to the existing traditional tourist recommendation system we aim at providing better recommendation along with a user friendly interface.

The travel industry, these days includes mass accessibility and mass investment in occasions. However a traveler can't choose which spot to visit, or where to stay. Current systems recommend places based on trends around us and many times user's personal preference and interests do not match with the recommended location. There are systems that also use item based collaborative filtering. The recommendation in item based filtering becomes monotonous after certain period of time [5]. In user based recommendation system locations will be recommended based on and also user info there will be better recommendation as the user’s database gets bigger. We can make the recommendation systems more robust and overcome these issues, by using user based recommendation system as an add-on to traditional system.

II. EXISTING SYSTEM

Traditional tourist recommendation system prominently uses collaborative filtering. There are two types of collaborative filtering item-item and user-user.

a) Item-Item based collaborative filtering

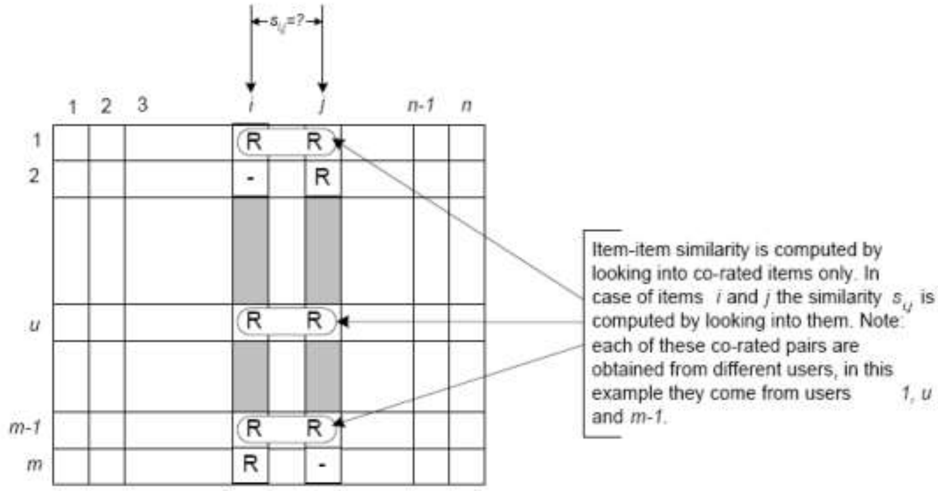
Item Based Collaborative Filtering (IBCF), suggests an item based on items the user has previously consumed. It looks for the items the user has consumed then it finds other items similar to consumed items and recommends accordingly. Item-based collaborative filtering is a model-based algorithm for making recommendations. How IBCF works is that it suggests an item supported by items the user has previously consumed. It looks for the things that user has consumed then it finds other items similar to the users taste and recommends accordingly [6]. The value between items is measured by observing all the users who have rated these items similarly. In the below diagram we can see the representation of the similarly rated items by the users.

IBCF is also known as Model-Based algorithm. The collaborative filtering process is envisioned as computing the expected value of a user prediction as defined earlier, given his/her ratings on other objects, by algorithms in this category. Different machine learning algorithms, such as Bayesian networks, clustering, and rule-based approaches, are used to construct models. The output interface in terms of prediction is the most critical step in a collaborative filtering scheme. Once we isolate the set of most similar items supported by the similarity measures, subsequent step is to analyze the target user’s ratings and use an appropriate method to obtain the predictions. Most commonly used methods are –

•Weighted Sum

•Regression

There are certain disadvantages in IBCF like the early rater problem. The early rater problem occurs when a new user is introduced to the system and is yet to rate a variety of items significantly large enough for the service to start suggesting similar items.



fig(1): item based recommendation

The algorithm calculates the similarities between different items in the dataset using one of several similarity methods, and then uses these similarity values to predict ratings for user-item pairs that are not from the dataset. The similarity values between objects are calculated by looking at all of the users who have rated them. The extent to which two objects are similar is determined by the ratings provided to them by users who have reviewed both of them. To measure the similarity between two items, a number of different mathematical formulations can be used. One of them is Cosine-based similarity. This formulation, also known as vector-based similarity, considers two items and their ratings as vectors, and determines similarity as the angle between these vectors:

$$sim(A_i, B_i) = \frac{\sum_{i=0}^{n} A_i \times B_i}{\sqrt{\sum_{i=0}^{n} A_i^2} \times \sqrt{\sum_{i=0}^{n} B_i^2}} \quad (1)$$

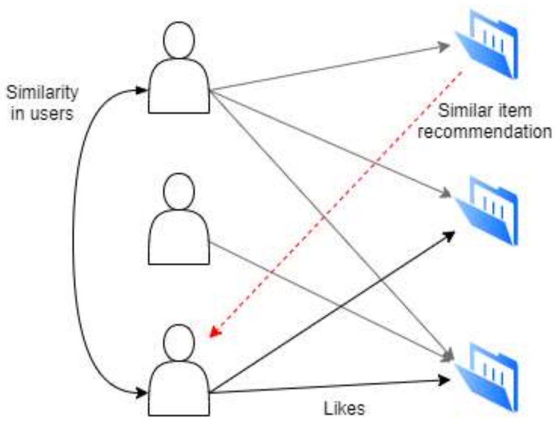
We can predict the rating for any user-item pair using the concept of weighted sum once we've created a model using one of the similarity measures mentioned above. We start by gathering all of the items that are close to our target item, then selecting items that have been rated by the active user. The similarity between each of these items and the target item is used to weight the user's rating for each of these items. Finally, to get a fair value for the expected ranking, we scale the prediction by the number of similarities:

$$P_{u,i} = \frac{\sum all\ similar\ items, N\ (s_{i,N} \times R_{u,N})}{\sum all\ similar\ items, N(|s_{i,N}|)} \quad (2)$$

One of the disadvantages of item based collaborative filtering is that if there are no locations similar to locations visited by user in the selected area then no new recommendation can be provided. Another disadvantage is that if a user has a travel history where he/she has visited similar type of locations then the recommendation provided will also be only based on that same type. For e.g., if the user has visited only restaurants which are famous for pizzas, then the traditional system will recommend other locations which are famous for pizzas which in long term makes the recommendations monotonous.

b) User-user collaborative filtering

User-Based Collaborative Filtering (UBCF) is a technique for predicting the items a user would like based on the ratings provided to that item by other users with similar tastes to the target user. UBCF is a form of collaborative filtering for recommender systems based on the similarity between users. The UBCF approach is based on the premise that if a person A and a person B have the same opinion on a topic, A is more likely to have B's perspective on a different subject than a random individual [1].



fig(2): user based recommendation

UBCF basically works on the following steps:

• A user's interests are expressed by rating objects (e.g., books, movies or CDs) of the system. These scores can be interpreted as a rough estimate of the user's interest in the corresponding domain.

•The system matches this user's ratings against other users' and finds the people with most" similar" tastes.

• For similar users, the framework suggests things that similar users have rated highly but that this user has not yet rated (presumably the absence of rating is often considered as the unfamiliarity of an item)

The advantages with this approach include:

• The results explain ability, which is an essential feature of recommendation systems.

•Easy creation, use and facilitation of new data.

•Content-independence of the items being recommended and good scaling with co-rated items.

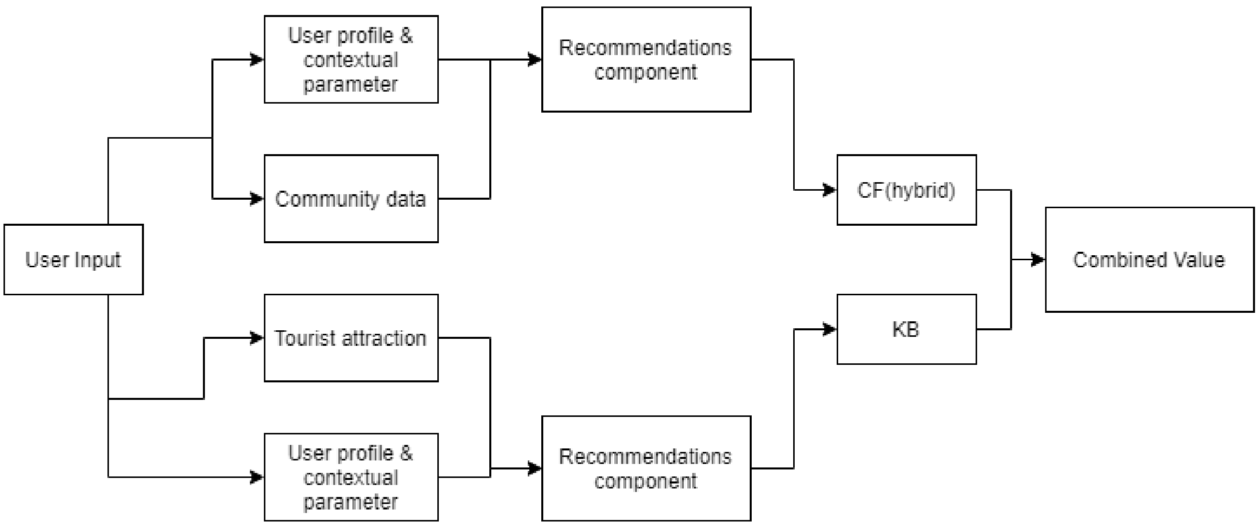
There are some disadvantages in UBCF, like Cold start, Shilling attack. In This approach if the database of users is small then there will be no recommendation or not that accurate as there would be no similarity between users this is known as Cold start problem. Also, when a person or group of people create multiple accounts in order to promote certain content and take away user's interests in other, in an attempt to promote their own products and hurt their competitors, and is an attempt to manipulate users into buying, watching or subscribing to a certain type of content based on a hidden agenda this is known as Shilling attack. Both the methods of Collaborative Filtering (CF) have certain advantages and disadvantages, combining both approaches we will be making an improved Hybrid system which will be pro-viding better recommendation.

III. IMPLEMENTATION METHODOLOGY

Our proposed solution will be a hybrid recommendation system, which will guide the user in field of tourism. We will be using phone application so that user needs not to access the web to search, book a hotel, restaurant or tourist place.

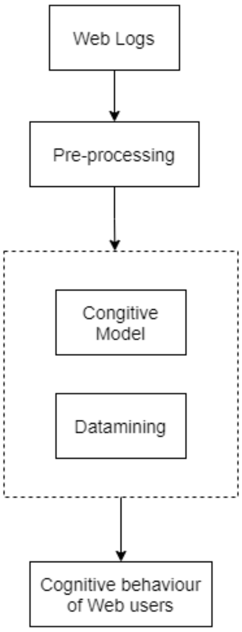
After logging into the application user can search in a particular category based on location filter. The categories in which user can search are restaurants, hotels and tourist spots. If a user is new to the application he/she will have to fill the sign up page with a basic survey form. Various tags/fields will be asked from user in all categories like cuisines for restaurants, user rating preferences. User's travel history will be taken for user based recommendation. When a user uses application, the user will be asked for the location in which user wants to search in. After selecting the category based on item filters and travel history taken during sign up will be used for creating one recommendation list (item based collaborative filtering recommendation list).

The second recommendation list will be generated using user-user based collaborative filtering. The users will be classified using k-NN classifier; the classification would be done on the bases of similar users [2]. User's similarity is based on distance matrix calculated using primary factors defined in the project. Different distance formulae like Euclidean Distance Formula, Cosine Similarity, Hamming Distance and some other are used to calculate the distances out of which Euclidean distance gives more accurate result in k-NN classification [3]. The major drawback of user-user based collaborative filtering recommendation is that due to less number of users there is problem of cold start. However we are making hybrid system, in initial stage when there is less number of users the recommendation list will not be affected as more preference is given to item-item based collaborative filtering recommendation list.



fig(3): hybrid recommendation system block diagram

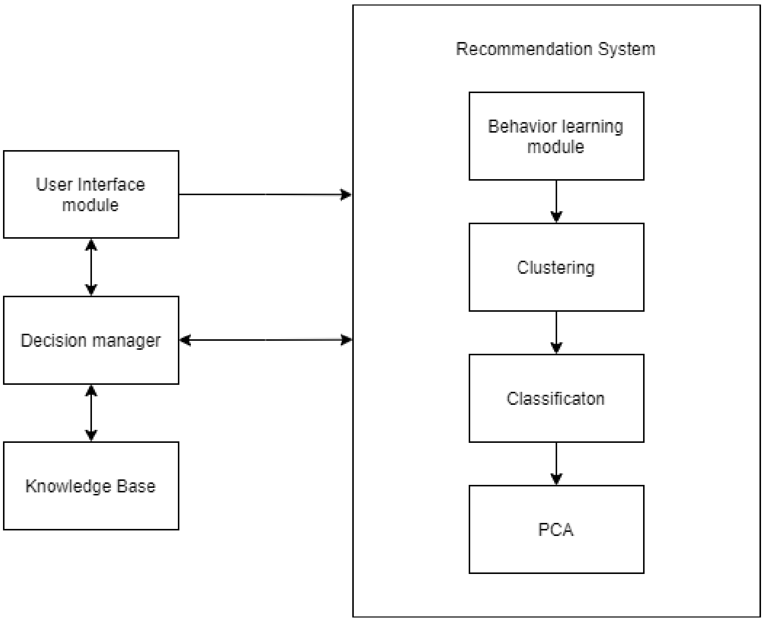
By getting a hybrid recommendation list we will enhance the recommendation [8], also by adding click stream analysis we can understand the user's behavior and improve the recommendation model [4]. In click stream we can consider a variable of place that user has searched in recent times, based on that analysis we will be able to suggest places user can visit in particular time. The variable can be varied based on restaurants or tourist places to visit. Further we can use this analysis to provide the user with certain offers and interesting deals. Click stream analysis has always been used to understand user's foreign behavior. The foreign behavior of the user can be analyzed using click-stream in the following way:



fig(4): click-stream methodology

The proposed paradigm (Fig. 4) combines data mining with the ACT-R (adaptive regulation of thought-rational) conceptual framework of cognitive architecture to provide cognitive behaviors of online customers.

We aim to use this analysis as an internal feature. Let us consider the following situation, where multiple users have booked a hotel after viewing 3rd image. Considering this response we can understand that we can set 3rd image as primary image for particular hotel. Thus not only making application more user friendly but also helping in the sales of hotel. The internal analysis of user's behavior will not only be restricted to viewing the images but variables can be extended to reviews, ratings, booking history etc. The parameters of the variables can be varied depending on how thorough application needs the analysis. Another algorithm used for internal analysis of user behavior.



fig(5): user behavior analysis

We can add an additional feature to recommendation system where we will recommend the user to visit popular locations as shown in fig (5). The popularity of a place is usually determined by the number of visitors, the type of tourist, and the amount of time spent there. We combined these features with our expanded collection of features to gain a better understanding of successful popularity patterns. All features are incorporated for under-emphasized locations to find the famous spot, even if it did not receive enough promotions. Though there are many factors that influence a location's popularity, not all of them are equally important. In addition, as the size of the explanatory variables grows, the likelihood of the model over fitting grows. As a result, dimensionality reduction is achieved by using principle component analysis (PCA) [7].

The steps of the PCA Algorithm are given below:

- We begin with data set 'A,' which is represented as a m x n matrix, with m rows representing variables and n columns representing samples, i.e. factors. This matrix will now be linearly transformed into another matrix 'B' of the same dimension m x n, so that for some matrix Z given by equation (3).

$$B = Z \times A \tag{3}$$

- Normalization is a crucial step in the algorithm, since it requires calculating the mean of the original data matrix and subtracting it from the mean in order to find the principal components, as shown in equation (4).

$$Mean(M) = \frac{1}{N} \sum_{n=0}^{N} (A)[\mathrm{m}, \mathrm{n}] \tag{4}$$

- In equation (5), compute A's covariance matrix, which will be of dimension m x m. Each element of the covariance matrix C_A represents all possible covariance pairs. In fact, all of the matrix's diagonal elements represent variance, while the non-diagonal elements represent covariance.

$$C_A = A \times \frac{A^T}{(n-1)} \tag{5}$$

- We must choose certain characteristics of the transformed matrix 'B' that correspond to the characteristics of the corresponding covariance matrix C_B. Since small variance data can be redundant, we must optimise variance while minimising covariance.
- As a result, we must choose a Z value such that the covariance matrix, C_B, becomes a diagonal matrix.

$$C_B = B \times \frac{B^T}{(n-1)} = \frac{(ZA)(ZA^T)}{(n-1)} = \frac{(ZA)(A^T \times Z^T)}{(n-1)} \tag{6}$$

$$C_B = \frac{Z(AA^T)Z^T}{(n-1)} = \frac{ZY \times Z^T}{(n-1)} \tag{7}$$

- The matrix Y can be represented in the form given in equation (8). Since any square matrix can be diagonalized orthogonally.

$$Y = EDE \tag{8}$$

- D is a diagonal matrix with the Eigen values of Y as its (diagonal) entries, and E is a m × m orthonormal matrix whose columns represent orthonormal Eigen vectors of Y.

- The value of the transformation matrix Z is determined here. The Eigen vectors of Y are made to be the rows of the Z. As a result, in equation (9),

$$Z = E^T \tag{9}$$

- After substituting Z value

$$C_B = \frac{ZY \times Z^T}{(n-1)} = \frac{E^T(ED \times E^T)E}{(n-1)} \tag{10}$$

$$C_B = \frac{D}{(n-1)} \tag{11}$$

Since the largest Eigen value indicates the relative significance of the corresponding principal variable, the Eigen values are arranged in descending order. The Eigen vectors in matrix E must be arranged in the diagonal matrix given in matrix according to their respective Eigen values.

IV. Future Scope

We have initiated the journey of developing the hybrid recommendation system discussed in the paper. It will help serving country travel market offering great help to the tourist with suitable recommendation. The application's growth in disposable income levels will drive future growth of Travel market not only in India but also other countries. Also, by implementing this hybrid system there will be a scope of improvisation in the current recommendation systems.

V. Conclusion

By implementing our proposed system we can overcome the drawbacks of item-item and user-user based collaborative filtering. The cold start problem faced during the initial phases when user data is less in user-user collaborative filtering, the item-item collaborative filtering will overcome this problem. And the monotonous result problem in item-item collaborative filtering is overcome by user-user collaborative filtering. Also by integrating click stream algorithm and PCA we can get advance analysis of user behavior, which can be used to enhance the recommendation.

VI. Acknowledgment

Mrs. Kavita Jain directed the creation of this project. Her extensive knowledge of Machine Learning and Computer Vision has given us a greater understanding of the application development process.

References

[1] Z. Jia, Y. Yang, W. Gao and X. Chen, "User-Based Collaborative Filtering for Tourist Attraction Recommendations," 2015 IEEE International Conference on Computational Intelligence & Communication Technology, 2015, pp. 22-25, doi: 10.1109/CICT.2015.20.

[2] Riswanto, E & Robi'in, Bambang & Suparyanto,. (2019). Mobile Recommendation System for Culinary Tourism Destination using KNN (K-nearest neighbor). Journal of Physics: Conference Series. 1201. 012039. 10.1088/1742-6596/1201/1/012039.

[3] Abu Alfeilat, H., Hassanat, A., Lasassmeh, O., Tarawneh, A., Alhasanat, M., Eyal Salman, H., and Prasath, V. 2019. Effects of Distance Measure Choice on K-Nearest Neighbor Classifier Performance: A Review. Big Data, 7(4), p.221-248.

[4] A. V. Bharathi, J. M. Rao and A. K. Tripathy, "Click Stream Analysis in E-Commerce Websites-a Framework," *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, 2018, pp. 1-5, doi: 10.1109/ICCUBEA.2018.8697475.

[5] Comparison of User Based and Item Based Collaborative Filtering Recommendation Services, PETER BOSTRÖM, MELKER FILIPSSON

[6] Nitin Pradeep Kumar, Zhenzhen Fan, Hybrid User-Item Based Collaborative Filtering, Procedia Computer Science, Volume 60,2015,Pages 1453-1461,ISSN 1877-0509, https://doi.org/10.1016/j.procs.2015.08.222.

[7] Ranak Roy Chowdhury, Muhammad Abdullah Adnan, and Rajesh K. Gupta. 2020. Real-Time Principal Component Analysis. ACM/IMS Trans. Data Sci. 1, 2, Article 10 (July 2020), 36 pages. DOI:https://doi.org/10.1145/3374750

[8] S. Wairegi, W. Mwangi and R. Rimiru, "A Framework for Items Recommendation System Using Hybrid Approach," 2020 IST-Africa Conference (IST-Africa), 2020, pp. 1-15.