Recommendation System for SmartMart-A Virtual Supermarket

Poonam Ghuli[*], Manoj Kartik R, Mohammed Amaan, Mridul Mohta, N Kruthik Bhushan, Poonam Ghuli, Shobha G

Department of Computer Science and Engineering, R V College of Engineering, Bengaluru,560059, India

A R T I C L E I N F O

A B S T R A C T

The current online shopping system does not provide the actual shopping experience to the users where they scroll through the web pages and select an item to purchase, which becomes monotonous soon and lacks the shopping experience. This paper describes the development of the SmartMart, a 3D shopping mart where the user can navigate around the mart with the help of a virtual cart and interact with the products as they would do in real life. Further, this paper explains the integration of the recommendation system with the SmartMart that provides efficient and seamless customer experience. Here, the customers are recommended with various products based on their previous purchase history. This gives a more personalized touch to each user adding on to their shopping experience. This paper describes the implementation of recommendation system using three types of algorithms namely item-based collaborative filtering, popularity model and user-based collaborative filtering. These algorithms are integrated with Unity SmartMart application and tested with the open source dataset. This dataset is organized in three ways-original data, data with a dummy field and normalized data. Distance metrics such as Pearson co-relation co-efficient, Jaccard similarity, cosine similarity metrics are used to compute the similarity among customers. This paper presents a comparative study of these different similarity metrics for user-based recommendation algorithm based on precision, recall, F-value and accuracy. From results obtained it can be seen that the Pearson co-efficient metric gives the highest accuracy value (86.6%) but the F-value, mean precision and mean recall values are very less compared to Jaccard Similarity Metric. Hence, Jaccard similarity metric is preferred over Pearson co-relation co-efficient. The design and development of SmartMart along with the recommendation system adds a new dimension to both the existing online shopping system and provides better customer satisfaction.

1. Introduction

3D simulation software provides a rich training experience to the user for various end-user applications, without using any physical asset and loss to the company. This is a key factor of 3D simulation that is expected to boost market growth. The simulation software market was valued at USD 8.24 billion in 2019, and it is expected to reach USD 19.22 billion, by 2025. The 3D simulation enables users to immerse in a computer-generated environment where they can interact with various simulated objects in a way replicating the real-life process. These 3D objects can be generated automatically or manually created by deformation of the mesh structure, or otherwise manipulating vertices. Using 3D simulation

for online shopping primarily enhances the customer experience and takes online shopping to a new level of sublimity.

This adds a realistic window which is absent with the existing online shopping websites where the user does not get to perceive the actual depiction of an item which they are inclined to purchase. Though the current websites try to minimize the hassle of the actual shopping process but it cuts away many privileges of a customer which would be helpful in selecting the appropriate item they want to purchase.

Currently, the e-commerce websites display 2D images of the products and facilitates the text-based search to seek out desired items to be purchased. But this does not allow customers to associate themselves with the real-world object experience. So, to replicate the real-world shopping experience for customers, this paper describes the design and development of a 3D shopping mart

[*]Corresponding Author: Poonam Ghuli, Department of Computer Science and Engineering, RV College of Engineering, 9986421230 & Email: poonamghuli@rvce.edu.in

called SmartMart. Here, each object is modeled in three dimensions to recreate the real-world object and also includes the other aspects of physics such as rotation, rolling, dragging of objects that makes the whole process more engaging. Due to this, the customer's experience of selecting an item from the SmartMart becomes more impeccable.

The SmartMart is designed using Unity along with its copious features to establish a smooth user interface. Also, the products to be bought are modeled and rendered in the gaming engine in way that provides the customers with a flawless online shopping experience. MySQL database used stores the information about the products, customers and their transaction details. MySQL database further enables to scale the number of products or customers without any inconvenience.

To broaden the scope of this proposed idea, a recommendation system has been implemented. To recommend appropriate products to customers the purchase history of multiple users is considered. A few factors that measures E-commerce success and customer satisfaction are as follows:

- What are the most trending products for sale?

- What are most suitable products for a particular customer based on his purchase history?

- What items are more often bought together?

Considering all the above-mentioned key factors, this paper describes implementation of recommendation system using three types of algorithms namely item-based collaborative filtering, popularity model and user-based collaborative filtering. Further, these algorithms are designed and integrated with Unity SmartMart application. For this, an open source dataset is used and organized in three ways- original data, data with a dummy field and normalized data. Further, Pearson co-relation co-efficient, Jaccard similarity, cosine similarity matrices are used to compute the similarity between purchase pattern usages among customers. Jaccard similarity metric provided the best result.

The admin of the SmartMart application is authorized to append the data of the ongoing transactions and periodically train the model with latest dataset. This ensures that the recommendation system takes into account the changing trends of customer's shopping patterns with proposed SmartMart application.

2. Literature Review

In [1], the author described a personalized Recommender System based on data clustering techniques to deal with the scalability problem. User-clusters are formed to categorize them based on their shopping behavior. The proposed work use DBSCAN clustering algorithm for clustering the users, and then implement voting algorithms to recommend items to the user depending on the cluster into which it belongs. Their idea was to partition the users of the Recommendation system using clustering algorithm and apply the Recommendation Algorithm separately to each partition. The system recommends item to a user in a specific cluster only using the rating statistics of the other users of that cluster. They have tested the algorithm on the Netflix prize dataset.

The work in [2], proposed two recommendation models for offline grocery/retail stores. The accumulated PoS (Point of Sales)

data is used to understand the shopping behavior of different customers and recommend better products. The proposed recommendation models are-Direct recommendation of products and two step recommendation systems. These recommendation algorithms are analyzed based on precision and accuracy of the model.

In [3], the author presented a recommendation system in their 3D environment where they consider the pre-purchase ratings of users to suggest products. The pre-purchase ratings include the interaction of user with the 3D objects. The user's emotions are captured in the form of electroencephalogram (EEG) signals. This makes the recommendations more users specific and accurate. A combination of this and the already existing recommendation system makes it dynamic.

The study in [4] presents a Recommendation system which offers real-time personalized suggestions on offers to its customers. Using iBeacons personalized notifications telling about the offers which are pushed to the smart phones of the customer in real-time as they navigate through the shopping mart. This approach uses a Bayesian Inference and a Hard k-modes clustering approaches. Bayesian Inference approach uses Entropy-based algorithm to give accurate and user specific recommendations on the offers provided to each user.

In [5], the author has proposed a virtual shopping mart which simulates a real working of shopping mart. The virtual shopping mart enables a cost efficient and high flexible way to store data. Researchers can use this data to improve the shopping experience of the customers. The data can be used to accurately predict the reactions of the customers to price change and the reactions to availability of the products that the customers need. These reactions can be used to optimize the shopping mart in a way to efficiently serve the customer's requirements.

In virtual reality the customers have the exact 3D representation of the items unlike the other 2D shopping websites this helps in analyzing the exact reactions the customers have towards a product in the shopping mart which enables researchers to capture an accurate consumer behavior.

The architecture that is proposed in this paper stands out from the other existing prototypes in a way that it integrates the recommendation system with the 3D environment (Unity software) which makes it possible for the user to have a sublime shopping experience. This entire system is based on client server model that allows concurrent access to the SmartMart by the potential customers. The use of recommendation system which recommends the most popular items and the items that the user would purchase helps the user to manage his/her shopping time more efficiently. The interaction with the 3D environment using keyboard and mouse makes the customer's shopping experience realistic.

3. Proposed Design of Smart-Mart

The system architecture for the proposed SmartMart is illustrated in Figure 1. It manifests a computing model in which a server hosts, delivers and manages most of the resources and services to be utilized by the clients. The project aims at building a Client-Server architecture where the client application entails a 3D UI (User Interface) platform that would grant customers/users a real-time shopping experience at the click of a button.

The customer interacts with the environment through keyboard and mouse input controls to bring in various changes ranging from the state of a product to updating the fields in database in the server. The MySQL database stores all the information about products, customers, transaction details and billing information. A virtual shopping cart designed in unity platform can freely move around the SmartMart.

Also, customers are allowed to add products to shopping cart, thereby facilitating a user-friendly experience to the users, unlike the existing shopping platforms. As the user proceeds towards the billing section, the inventory is updated which would help administrators to track and maintain the stock efficiently.
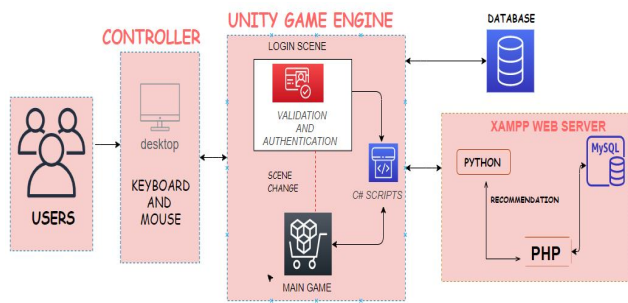

Figure 1: System Design of User interaction with SmartMart.

The Figure 1 describes about the Unity gaming engine that is used to host the entire shopping environment application. The Unity APIs used in the proposed work handle various user interactions and allow communication with third party scripts, namely, C# and PHP (Hypertext Preprocessor). The user interactions are translated into corresponding events by the interface. The application processes these events and triggers other events or updates the SQL database connected to it. A PHP script acts as an intermediary to communicate between the SQL database and the application.

The figure 2 describes the 3D view of the proposed SmartMart. This figure shows a virtual cart an products placed on shelves in the SmartMart. The user can move around virtual around the mart along with the Virtual-Cart.

The uniqueness of this paper is integration of SmartMart with the recommendation system. This recommendation system is deployed using the Python script and connected to the SmartMart application via PHP.

Figure 2. The 3D view of the SmartMart

4. Workflow of Proposed Work

Unity is a cross-platform game engine with a built-in IDE (Integrated Development Environment) used to develop video games for web plug-ins, desktop platforms, consoles and mobile devices. Pro-Builder is a tool that gives Unity the ability to edit the shape of primitive objects and model these 3D objects as per the requirements.
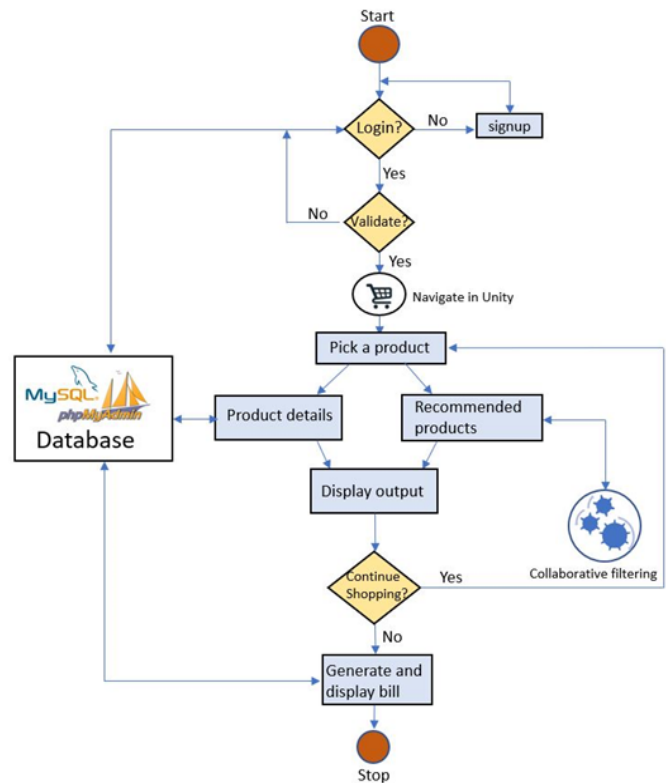

Figure 3: Workflow of Smart-Mart

The work flow of the Virtual Supermarket is depicted in the Figure 3. The application starts with the initial scene where user credentials are validated with username and password. If the credentials are correct, the user gets into the game scene, which is the main application. The user is placed at the entrance of the mart, with a virtual cart placed there. The cart is designed using Pro-Builder. The cart is scripted such that the customer can move it around the mart with the help of the mouse control.

Shelves at the virtual mart are rendered using unity platform and various products are placed on it. The products at the mart are built as 3D objects using Pro-builder platform in order to recreate the real-life products such as baseball bats or celery. These virtual products created in the Unity platform are nothing but Game Objects. Game Objects are the fundamental objects in Unity that represent characters, props and scenery. Every Game Object is associated with a tag which is used to identify, retrieve and store the information about a particular product. Each product/Game Object is associated with a script which allows the user to rotate, drag and drop it into the virtual cart. Whether the customer puts or removes the item to or from the cart, the inventory database is updated accordingly in real time. As soon as the user selects the product, the details of the product along with the recommendations are displayed on the monitor installed at the billing counter. This is accomplished using PHP server that communicates with MySQL and Unity.

Once the user completes the adding of products to the virtual cart, the cart can be dragged to the billing counter where appropriate billing information is displayed. This billing information includes the items purchased, per piece price, total quantity, total price, time of billing, customer id and customer details. These transaction details are stored in the database in real time. Once the billing details are generated, the user signs out and the session is over.

Thus, database manages all the transactions, user and product information efficiently. The details of the products are displayed to customers as required without any latency. The interactions, ranging from dragging and dropping of the items into virtual cart to its movement in 3D environment in the first-person view is flawless.

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper; use the scroll down window on the left of the MS Word Formatting toolbar.

5. Recommendation System

Al the proposed virtual Super market is further integrated with a Recommendation System. In the field of e-commerce, recommendation systems are an attempt to bridge the gap between the online and in-store shopping experience reiterating the very aspect that made this project distinctive. Recommendation systems streamline our online shopping experience by suggesting products that customers might be interested in.

With the growing technological advancements in the fields of business intelligence and data science, recommendation systems have become increasingly popular. They help the customers to know and decide what to buy based on their previous purchase history, their feedbacks and features of the products to be purchased.
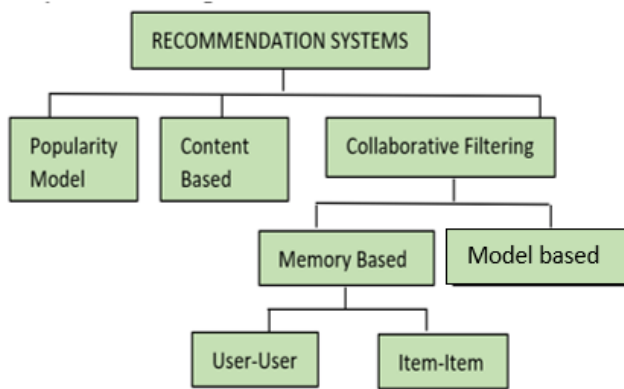

Figure 4: Classification of recommendation system

The two most popular approaches for building a recommendation system are Collaborative Filtering and Content-based algorithms as shown in Figure 4. While the former approach

requires only the user's historical preferences over the existing inventory, the latter requires a fair amount of knowledge on the item's characteristics such as ingredients, trademark and features. Although Collaborative Filtering depends on only purchase history of users one assumption that this model makes is that the users who have purchased a certain set of products in the past also tend to agree with the same kind of products in the future.

Collaborative filtering algorithms [6] are mainly classified as model based and memory based approaches. Further there are two types of Memory based Collaborative filtering algorithms namely Item-based collaborative filtering and User-based collaborative filtering . Item-based collaborative filtering that depends on items and their similarity to other items. User-based collaborative filtering algorithm is more resource-intensive as it looks at customer behaviors and preferences. In memory based approach the similarities with the closest users or items are calculated by using Cosine similarity or Pearson correlation coefficients, which depend primarily on arithmetic operations. Model based collaborative filtering algorithms provide recommendations by developing a machine learning model on existing user ratings and then to predict user's rating of unrated items [7]. As building a model is both time and resource consuming. It is difficult to add data to model based systems. The quality of prediction for memory based algorithms is good as it uses the entire database every time it makes a prediction. Thus, we go with memory based algorithms for the proposed work.

The similarity metrics [8] used for recommending products in the SmartMart application are Cosine, Pearson Correlation Coefficient and Jaccard similarity methods.

- By measuring cosine of the angle between any two vectors, Cosine similarity technique calculates how these are related to each other. The Cosine similarity metric is computed as shown in equation 1.

$$CS(i,j) = \frac{\sum_{u \in U_{ij}} r_{ui} r_{uj}}{\left(\sqrt{\sum_{u \in U_{ij}} r_{ui}^2}\right)\left(\sqrt{\sum_{u \in U_{ij}} r_{uj}^2}\right)} \quad (1)$$

  where Ui represents users, who have purchased only item i & Uij represents users who have purchased both i & j items.

- Pearson coefficient similarity technique calculates the similarity between two vectors by the linear correlation between them. The values resulting from PCC range from -1, representing negative correlation to +1, representing a high positive correlation. Zero order correlation is when the value is 0 and the vectors do not show any relation. The Pearson Correlation Coefficient similarity metric is computed as shown in equation 2.

$$PS(i,j) = \frac{\sum_{u \in U_{ij}} (r_{ui} - \overrightarrow{r_i})(r_{uj} - \overrightarrow{r_j})}{\left(\sqrt{\sum_{u \in U_{ij}} (r_{ui} - \overrightarrow{r_i})^2}\right)\left(\sqrt{\sum_{u \in U_{ij}} (r_{uj} - \overrightarrow{r_j})^2}\right)} \quad (2)$$

  where Ui represents users, who have purchased only item i & Uij represents users who have purchased both i & j item.

- Jaccard similarity technique calculates the extent of similarity between two users by taking number of preferences common between them and is shown in equation 3.

$$JS(i,j) = \frac{|U_i \cap U_i|}{|U_i \cup U_i|} \qquad (3)$$

where Ui represents users, who have purchased only item i.

As mentioned in the previous section this paper implements recommendation system using three types of algorithms namely item-based collaborative filtering, popularity model and user-based collaborative filtering. Popularity based recommendation systems learn data in a certain way, after which, suggest items that are in trend at the moment. For instance, if there exists a product that is being purchased by many users then it is likely that it will be recommended to every user.

6. Details of Dataset and Data Pre-processing

The proposed work makes use of two datasets. First one is MySQL database created to store details pertaining to the customers, their transactions and product information. The other one is an open-source dataset [9].

The MySQL database being employed has the following tables that stores vital information – the validation details of the customers; the customer table contains the details of all the existing customers; the inventory table contains details like product ID, product name and their respective quantities that directly correlate to the stock available with the seller; the billing table which stores the bill generated during shopping and holds information like user ID, items bought, time of purchase, etc and the orders table that contains all the bills generated along with respective product information.

The open-source dataset that is used to build a recommendation system is available in .csv format. It has information regarding the purchase history of the customers. There are 3 approaches in terms of how data is being used.

Taking customer ID and product ID as they are.

Creating a dummy field for marking whether a customer bought an item or not.

Normalize purchase frequency of each item across users by first creating a user-item matrix.

Since normalization is always used as a basic component for the predictor models and most often produces the best results when it comes to recommender systems, it was decided to proceed with normalized data. We have normalized the purchase history of users on a scale of 0 to 1. 1 depicts the greatest number of purchases for an item and 0 depicts 0 purchase count for that item.

7. Results and Discussions

The proposed work evaluates the different Similarity Metrics for user-item recommendation algorithm based on recall, precision, and F-value with the number of recommended items for each user as 10 [10]. The results presented in Table 1 evaluate the different similarity matrices for user-item recommendation algorithm by using the average value of recall, precision, and F-value. Table 1 also provides the recall, precision, and F-value for popularity model variant of recommendation system. Cut-off 'K' is a user definable integer that is set by the user to match the top-N recommendations. The value of cut-off is kept as 10 (K=10).

Table 1: Performance comparison of Similarity Matrices

Recommendation System Models	Similarity Metrics	Precision	Recall	F-Value
Popularity Model	-	0.002685	0.014068	0.004509
User-based Collaborative Filtering Algorithm	Cosine similarity	0.026191	0.137444	0.043999
	Pearson similarity	0.002706	0.014176	0.004545
	Jaccard similarity	0.027089	0.141439	0.045469

Precision: The number of products bought by the customer out of all the recommended products. Precision at k is the proportion of recommended items in the top-k set that are relevant. Suppose if 10 products were recommended to this customer out of which he buys 5 products, then precision is 0.5. The precision at k is computed using equation 4.

$$\text{Precision} = \frac{\left(\begin{array}{c}\text{Number of recommended items that} \\ \text{are relevant in the top k set}\end{array}\right)}{\text{Number of recommended items in the top k set}} \qquad (4)$$

Recall: The number of products that are recommended out of all the products that a user buys. Recall at k is the proportion of relevant items found in the top-k recommendations. Suppose if a customer buys 10 products and the recommendation decided to show 4 of them, then recall is 0.4. Recall at k and F - value is computed as shown in equation 5 and equation 6 respectively.

$$\text{Recall at k} = \frac{\left(\begin{array}{c}\text{Number of recommended items} \\ \text{that are relevant in the top k set}\end{array}\right)}{\text{Total number of relevant items}} \qquad (5)$$

$$\text{F} - \text{value} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \qquad (6)$$

From Table-1 it can be inferred that the Jaccard similarity has the highest F-Value (0.04546) followed by cosine similarity (0.04399). The mean precision and mean recall are also the highest for Jaccard similarity (0.0271 and 0.1414 respectively). By the results it can be seen that Jaccard similarity is the best metrics to recommend products for the dataset used in this paper.

Table 2 presents the performance comparison of various similarity metrics based on RMSE (Root Mean Square Error) and accuracy. It also provides RMSE and accuracy values for the

popularity model variant of recommendation system. The Pearson metric gives the highest value (86.6%) but since the F-value, mean precision and mean recall values are very less, Jaccard metric is preferred. Note that the popularity model just recommends the most popular products and is similar for every user, so we don't consider it in user-item recommendation.

Table 2: Performance comparison of various similarity metrics based on RMSE and Accuracy

Recommendation System Models	Similarity Metrics	RMSE	Accuracy
Popularity model	-	13.3580%	86.6420%
User-based Collaborative Filtering Algorithm	Cosine similarity	16.1986%	83.8014%
	Pearson similarity	13.3915%	86.6085%
	Jaccard similarity	16.0924%	83.9076%

8. Conclusion

The Virtual SmartMart application is an innovative platform, which evolves the way people shop online when compared to different E-commerce websites. The above deployed recommendation models make sure that the customers never miss out on any current trending products (Popularity model), the most liked products by other users (user-item model) and best products related to the items present in the cart (item-based model). Therefore, the recommended products will provide customers a sense of satisfaction.

This paper presents a comparative study of different similarity metrics for user-based recommendation algorithm based on precision, recall, F-value and accuracy. From results obtained it can be seen that the Pearson co-efficient metric gives the highest accuracy value (86.6%) but since the F-value, mean precision and mean recall values are very less, Jaccard similarity metric is preferred.

From the dealer point of view, Unity Analytics (integrated with Unity game engine) with SmartMart will help him in monitoring the most selling products, the most active time during a day, the number of shoppers active at any point of time and the countries with leading number of shoppers. This helps dealer make appropriate decision at right time. Integration of Unity Analytics with the SmartMart can be considered as future work for this work proposed here.

For any company retaining customers and up scaling of business are two very important aspects. SmartMart Unity application successfully helps a company in doing so. The limitations of the virtual supermarket include cold start problem, data sparsity problem and scalability. This can be overcome by developing a hybrid recommendation system; a combination of collaborative and content-based filtering. This can be taken up as future work.

Conflict of Interest

The authors declare no conflict of interest.

References

[1] J. Das, P. Mukherjee, S. Majumder, P. Gupta, "Clustering-based recommender system using principles of voting theory," in Proceedings of 2014 International Conference on Contemporary Computing and Informatics, IC3I 2014, 2014, doi:10.1109/IC3I.2014.7019655.

[2] N. Sano, N. Machino, K. Yada, T. Suzuki, "Recommendation system for grocery store considering data sparsity," in Procedia Computer Science, 2015, doi:10.1016/j.procs.2015.08.216.

[3] G. Guo, M. Elgendi, "A New Recommender System for 3D E-Commerce: An EEG Based Approach," Journal of Advanced Management Science, 1(1), 61-65, 2013, doi:10.12720/joams.1.1.61-65.

[4] P. Christodoulou, K. Christodoulou, A.S. Andreou, "A real-Time targeted recommender system for supermarkets," in ICEIS 2017 - Proceedings of the 19th International Conference on Enterprise Information Systems, 2017, doi:10.5220/0006309907030712.

[5] W.E. Waterlander, M. Scarpa, D. Lentz, I.H.M. Steenhuis, "The virtual supermarket: An innovative research tool to study consumer food purchasing behaviour," BMC Public Health, 11, Article No. 589, 2011, doi:10.1186/1471-2458-11-589.

[6] G. Suganeshwari, S.P. Syed Ibrahim, "A survey on collaborative filtering based recommendation system," in Smart Innovation, Systems and Technologies, 503-518, 2016, doi:10.1007/978-3-319-30348-2_42.

[7] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, "Item-based collaborative filtering recommendation algorithms," in Proceedings of the 10th International Conference on World Wide Web, WWW 2001, 2001, doi:10.1145/371920.372071.

[8] A. Agarwal, M. Chauhan, "Similarity Measures used in Recommender Systems: A Study," Minakshi Chauhan International Journal of Engineering Technology Science and Research IJETSR Www.Ijetsr.Com ISSN, 4(6), 619-626, 2017.

[9] S. Mohtaj, F. Tavakkoli, "Maglet: A Persian Journal Recommender System," in 9th International Symposium on Telecommunication: With Emphasis on Information and Communication Technology, IST 2018, 2019, doi:10.1109/ISTEL.2018.8660987.

[10] A. Gunawardana, G. Shani, "A survey of accuracy evaluation metrics of recommendation tasks," Journal of Machine Learning Research, 10, 2935-2962, 2009.