INTERNATIONAL JOURNAL OF ADVANCE RESEARCH AND DEVELOPMENT

IJARnD

(Volume 5, Issue 3)
Available online at: www.ijarnd.com

Mobile game applications Recommendation System with item-based Collaborative Filtering

Khin Mar Cho[1], Mya Sandar Kyin[2]
[1]Lecturer, University of Computer Studies, Pyay, Myanmar
[2]Lecturer, University of Computer Studies, Taungoo, Myanmar

ABSTRACT

Recommender Systems are software techniques that are being widely used in many applications to suggest products, services, and items to potential users. The main purpose of Recommender Systems is to provide meaningful recommendations about the items or products to a collection of users for their interested items. There are two popular approaches in recommendation: user-based and item-based collaborative filtering. The difference between them is that user-based takes users' behaviors and item-based takes items' rating values. The purpose of this paper is to present a recommender system that provides meaningful recommended mobile phone applications to the mobile phone users which are relative to their needs or targets. This system emphasizes mainly on item-based collaborative filtering method that bases on rating values of the items because the computational complexity of user-based recommendation grows linearly with the number of users. By using this system, the mobile phone applications users can obtain the optimized suggestions without their waste of time and effort.

Keywords— *Recommender Systems, Collaborative Filtering (CF), Item-Based, Rating Values*

1. INTRODUCTION

The World Wide Web (WWW) attracted the attention of researchers from many areas such as artificial intelligence (AI), natural language processing (NLP), information extraction, question answering, intelligent agents, ontology building for the semantic web and many others. The explosive growth of world wide web and the emergence of e-commerce have led to development of recommender systems [6]. Many commercial websites tried to upgrade their systems by using recommendation techniques and recommends their products or items. Recommender System (RS) is the information filtering that applies data analysis techniques to the problem of helping users and finds the products or items which they would like to purchase by producing a predicted likeness score or a list of recommended items.

RS is also a kind of an automated and sophisticated decision support system that is needed to provide a personalized solution in a brief form without going through a complicated search process. It learns from users and recommends products or items that they will find most valuable among the available items. It is being used by an ever-increasing number of ecommerce sites to help users who find products to purchase.

Information is growing exponentially over the internet. User gets confused while seeing so many items over the Internet to decide which one to buy. Recommendation techniques are important for many users who want to know which is latest or which is suitable for them. Typical recommender systems adopt a static view of the recommendation process and treat it as a prediction problem. Recommender systems employ prediction algorithms to provide users with items that match their interests [7]. The collaborative filtering became the most popular method for decreasing information conflicts and it worked like creating a database of preferences for users and items. The system has significant success on the Internet and most companies use collaborative filtering. The personalized information filtering technology used to identify a set of items that will be of interest to a certain user. Therefore, the user-based collaborative filtering is the most successful technology for building recommender systems to date and is extensively used in many commercial recommender systems.

Unfortunately, the computational complexity of these methods grows linearly with the number of customers, which in typical commercial applications can be several millions. To address these scalability concerns, many researchers has developed item-based recommendation technique. This technique analyzes the user-item matrix to discover relations between the different items and uses these relations to compute the list of recommendations. The purpose of item-based recommender system is to predict the rating that a user of the system would assign to an item and recommend to that user the items with the highest predicted ratings. The subsequent sections of this paper are organized as follows: The next section contains a brief overview of some related researches. In section 3, the background theory about the approach for collaborative filtering (CF), based on items, is described. The system design of proposed system is presented in section 4 and section 5 describes the performance evaluation of the system. Finally, the conclusions and future works are described in section 5.

2. RELATED WORK

Previous researches related to CF-based recommender systems, which can be divided into two classes: Memory-based CF and Model-based CF are briefly explained [1] in this section. The user-based approach has been the most widely used for recommendation systems since the first system to generate automated recommendations, the GroupLens [3], was proposed. User-based CF uses a similarity measurement between neighbors and the target users to learn and predict the preference towards new items or unrated products regarding a target user. Though user-based CF algorithms tend to produce more accurate recommendations, they have some serious problems relating to the complexity of computing each recommendation as the number of users and items grow.

In order to improve scalability and real-time performance in large applications, a variety of model-based recommendation techniques were developed [2, 3]. Especially, a new class of Item-based CF, which is one of model-based approaches and this research [4] focuses on, has been proposed. This approach provides item recommendations by first developing a model of user ratings. In comparison to user-based approaches, item-based CF is typically faster in terms of recommendation time, though the method may have an expensive learning or model building process.

Instead of computing the similarities between the users, item-based CF reviews a set of items the target user has rated and selects k most similar items, based on the similarities between the items. *Sarwar et al*. [2] evaluated various methods to compute similarity and approaches to limit the set of item-to-item similarities that must be considered. *Deshpande et al.* [5] proposed Item-based top-N recommendation algorithms that are similar to previous item-based schemes. They separated the algorithms two distinct parts for building a model of item-to-item similarities and deriving the top-N recommendations using this pre-computed model.

3. BACKGROUND THEORY

A Recommender System (RS) acts as a computer system that provides advice to users about items they might wish to purchase. It also provides individual personalization to each user by customizing its recommendations and presenting different items for each user wants. To provide a personalized set of recommendations, a recommender system incorporates a user's wishes into a user model and exploits suitable recommendation algorithms to map the user model into targeted item suggestions. There are three steps involved in a recommender system: acquiring preferences from a user's input data; computing the recommendation using proper techniques; and finally presenting the recommendation results to user. The goal of a recommender system is to generate meaningful recommendations to a collection of users for items or products that might interest them. Suggestions for books on Amazon, or movies on Netflix, are real-world examples of the operation of industry-strength recommender systems.

The design of such recommendation engines depends on the domain and the particular characteristics of the data available. For example, movie watchers on Netflix frequently provide ratings on scale of 1(disliked) to 5(liked). Such a data source records the quality of interactions between users and items. Additionally, the system may have access to user-specific and item-specific profile attributes such as demographics and product descriptions, respectively. Recommender systems differ in the way they analyze the data sources to develop notions of affinity between users and items, which can be used to identify well-matched pairs [6].

The recommender system can be differentiated by the ways in which they manipulate the available data sources to identify potential matches between users and items. Conventional wisdom about the taxonomy of techniques differ slightly from author to author, but encompass primarily collaborative filtering (user-user), content-based (item-item), knowledge-based (user-item), community-based (user-user), and hybrid based [7].

The recommender systems become important tools for internet marketing activities in ecommerce as they can provide a personal service for each user and support the user in product purchasing. They provide personalization on ecommerce sites by adapting product suggestions according to each user's preferences. The recommender systems also help ecommerce sites achieve mass customization by providing multiple choices of products that meet the multiple needs of multiple consumers. Many of the largest ecommerce websites such as Amazon.com, Apple Computer and Netflix DVD-rental are using recommender system approaches to assist their users in selecting products to purchase. In the Amazon.com site, for example, many types of recommendations are provided to assist users in making their downloading decisions. This site provides product suggestions to user based on applications that they have already rated.

In this proposed system, the recommendation list thus generated recommends applications that are similar to those in which the user has previously shown interest and hence, users are exposed to applications that they might have previously been unaware of in the uses liked feature, the system recommends applications frequently rated by other users who have rated the selected application.

3.1. Collaborative Filtering Recommender System

The collaborative filtering recommender system is the earlier and most successful recommendation technology. This approach is automatic and the "word-of-mouth" paradigm that evaluates items for the target user based on the opinion of other users. The main idea of this approach is that a target user is likely to enjoy the items that other users with common interests have liked. This approach assumes that human preferences are correlated, in that a user with similar asses will rate things similarly. Thus, explicit ratings are the typical input of this approach. The collaborative filtering approach only relies on collaborative knowledge sources such as collaborative opinion profiles, demographic profiles and user opinion and thus, it does not require other information about users or items. It can also be applied in many domains other than text-based items as in the content-based recommender systems. Collaborative filtering systems are implemented in various domains such as in the Usenet newsgroup articles domain: Grouplens, the applications domain: Amazon.com, and other domains [6].

Generally, the recommendation process consists of three steps, e.g, finding similar user, making neighborhood, and counting prediction based on selected neighbors. Collaborative filtering generated item predictions or item recommendations for targeted users. Item consists of interesting topics or thing such as books, films, arts, articles, applications or travel destinations. Ratings consisted of (a) scalar numerical value of integer; (b) binary value of boolean, agree or disagree, good or bad; (c) unary value indicated user history activity that the user has observed or purchased items or rated items. Unary value can be combined with binary value to provide user rating of positive or negative about product item rating value.

The availability of rating values indicates information connecting the user with the preferred items. Ratings can be collected explicitly, implicitly, or a combination between explicit and implicit. The explicit rating is obtained when the user is asked to provide an opinion on a particular item. Implicit rating is earned through the user intention. The unavailability of rating values will lead to the items are not recognized by machine learning and not displayed to users even though the product is existed in the database system.

The basic idea of collaborative filtering is that it provides item recommendation on the opinions of likeminded people. Collaborative filtering is used to find similarity in two forms. Prediction- it is a numerical value of the likeliness of occurrence of item. Recommendation- it is a list of items that the user will like the most. Collaborative Filtering Recommender System can be divided into two main approaches: User-based and Item-based [8]. However, this proposed system mainly emphasizes on the item-based collaborative filtering.

3.2. User Based Collaborative Filtering

User-based collaborative filtering approach tends to find similarity between users using similarity measures. Using the past usages and given rating to the items, similar users are calculated and the recommendations are generated depending on these sorted users. The user-based collaborative filtering takes the user's behavior and preferences. It uses statistical techniques to find a set of users, known as neighbors (neighbors) that have history agrees with the target user [9].

3.3. Item-Based Collaborative Filtering

Item-based collaborative filtering recommendation method bases on the similarities between giving ratings to the items that the users interested in them. Of the degree of similarity of items, and then divided by the parameter value of the user needs to acquire the item usability. Items that have highest utility value which then are made recommendations. This method appears as a solution to several problems in user-based collaborative filtering is the problems of limited and scalability as well as time and memory.

Item-based collaborative filtering method would know the value of similarity among the items with low rating level distribution and the value of similarity level distribution and the value of similarity among the items tends to be less likely to change than the value of similarity between users. Item-based collaborative filtering done by establishing a model similarity offline, which automatically would save time and memory used for the computation of the time the user accesses web pages for mobile phone applications provider [10].

Item-based Collaborative Filtering Recommendation System exhibits a more flexible trade-off between locality and scalability. System provides a more efficient way to maintain the user partitioning structure [11]. The item-based approach looks into the set of items that the target user rated and computes how similar they are to the target item. At the same time, the corresponding similarities {si1, si2,…….. , sij} are also computed. Once the most similar items are found, the prediction is then computed by taking a weighted average of the target user's ratings on these similar items.

In this type of algorithm, similarities between different items are calculated by using similarity measures and then these similarity measures are used to predict ratings. Item-based models use rating distributions per item, instead of per user. This leads to more stable rating distributions in the system. The following Figure 1 and Figure 2 give the illustration of user-based and item-based collaborative filtering respectively.

The system used computational similarity method between two items and find predicted items by counting the weighted sums of different item ratings on individual users. Item-based collaborative filtering contains recommendation algorithm based on similarity relationship between rated items and purchased items. From the level of item similarity, then they are divided by parameters of user needs to obtain product usability value. It is also so-called item-to-item collaborative filtering.

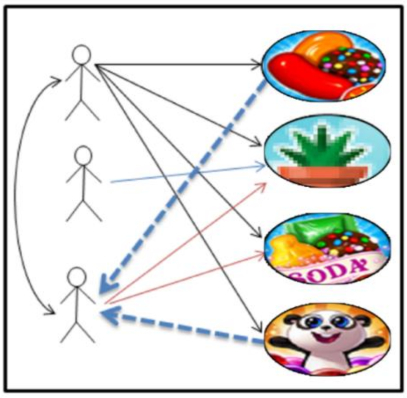

Fig. 1: User Based Collaborative Filtering

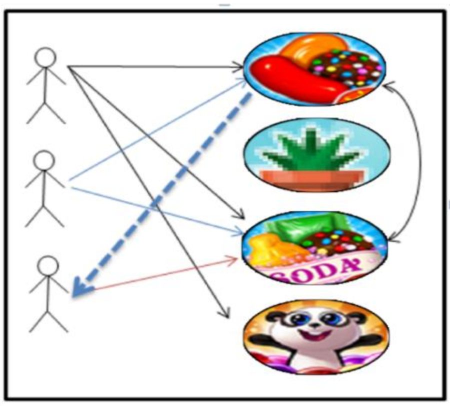

Fig. 2: Item Based Collaborative Filtering

3.4. Cosine Based Similarity

The cosine-based similarity and Pearson correlation are the most popular approaches to calculate the similarity between two users [6]. Cosine-based similarity works on the concept of statistical cosine where two items are considered as two vectors in the dimension m user space. [7] The similarity between them is measured by calculating the cosine angle between two vectors. For item list, the similarity between item i and j will form new direction and distance between the groups as represented by equation (2.1). Formally, similarity between items i and j, denoted by sim (i, j) is given by

$$sim(i,j)=\cos(i,j)= .(i.j)/(||i||2*||j||2) \tag{3.1}$$

Where "." denotes the dot-product of the two vectors.

It is a vector-space approach based on linear algebra than a statistical approach. The similarity between two items can be measured by treating each document as a vector of word frequencies and computing the cosine of the angle formed by the frequency vectors. This formalism can be adopted in collaborative filtering which uses users or items instead of documents and ratings instead of word frequencies [12].

3.4.1. Adjusted Cosine Similarity: The computation of similarity with basic cosine needs huge size of data which sometimes difficult for small dataset size. This case has one obvious drawback, and this needs modification for scoring scale among different users with small dataset size. The issue is then resolved with Adjusted Cosine Similarity approach as proposed by Chen. The similarity approach which using this scheme has a goal to spread the value between items with the level of small rating distribution. The Adjusted Cosine Similarity algorithm can modify the value of similarity between items. In addition, the algorithm also can estimate the frequent change of items and user relationship. It predicted similarities by forming an offline similarity model that automatically saves time and memory for counting when a user accesses a list of items. The popular similarity model which implemented in recommender systems is given in equation (3.2) [8]. The adjusted cosine similarity eliminates this drawback by subtracting the corresponding user average from each co-rate pair. The formulation for adjusted-cosine similarity is described below:

$$sim(i,j)=\frac{\sum_{u\in U}(R_{u,i}-\bar{R}_u)(R_{u,j}-\bar{R}_u)}{\sqrt{\sum_{u\in U}(R_{u,i}-\bar{R}_u)^2}\sqrt{\sum_{u\in U}(R_{u,j}-\bar{R}_u)^2}} \tag{3.2}$$

Where $R_{u,i}$ is rating of user u for item i,
R û is average rating of user u for all items,
$R_{u,j}$ is rating of user u for item j.

The adjusted-cosine similarity works in the following manner:

Step1. First, the algorithm accepts the product id, user id, ratings from the dataset table and transforms item to user-item matrix.

Step 2. In this step, the item-to-item matrix is computed. Here the actual adjusted-cosine similarity is implemented. It is calculated how similar an item is to another item.

Step 3. This step is concerned about finding the predicted rating for the items that user had not rated. This is also called as prediction computation. Weighted sum technique is used in this case [13].

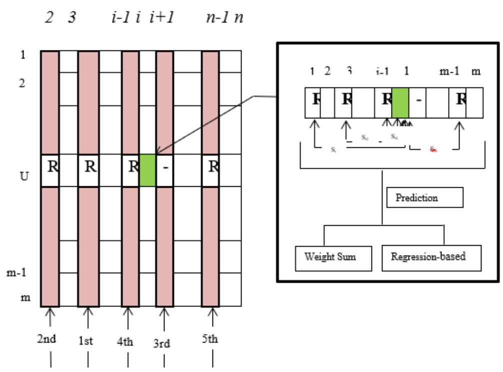

Fig. 3: Item Based Collaborative Filtering Algorithm

4. PROPOSED SYSTEM DESIGN

To avoid the scalability problem of user-based collaborative filtering method, another basic collaborative filtering algorithm called item-based collaborative filtering method is used in this system. This method analyzes the relationships between items rather than between users, because item relationships are relatively static. Similarity relations for items are computed offline. Item-based methods are also known as model-based methods. There are seven steps behind the process of the system. They are:

(1) Calculate Average Rating
(2) Consider the corresponding rating lists for the target rating
(3) Calculate adjusted cosine similarity of two items with respect to the corresponding rating lists.
(4) Generate the similarity matrix
(5) Find normalized rating
(6) Predict the target rating using weighted sum
(7) De-normalize the target rating.

The item-based collaborative filtering method makes recommendation according to the following simple step to step procedure:

- Users are requested to give numeric rating to the items.
- A recommender system correlates the ratings in order to determine which item's ratings are most similar to other item's ratings.

- The system predicts ratings of new items for the target user based on the ratings of similar items already rated by the users.
- If these new items seem to be preferred, the system recommends them to the user.
- Then, the user knows as predicted rating.

4.1. Computing Average Rating

The average rating is computed the total ratings is divided by number of applications.

$$\text{Average Rating } \overline{R}_u = \frac{\text{Total Ratings of User}}{\text{Number of Applications}} \qquad (4.1)$$

Table 4.1: Average Rating

No	Name	Candy Crush Soda Saga	Merge Plane	Angry Birds	Air Camera Photo Editor College Filter	Block Pzzle	Block Puzzle Conquer	Blossom Blast Saga	-	-	-	Jewels Jungle Match3 Puzzle	Average
		1	**2**	**3**	**4**	**5**	**6**	**7**	**-**	**-**	**-**	**42**	
1	U Myat Htun Kyaw	5	4	3	2	1	5	3	-	-	-	1	2.24
2	U Min Min	3	0	2	1	2	3	1	-	-	-	2	1.78
3	Daw Nwe Ni Win	5	4	3	2	1	2	2	-	-	-	3	2.52
4	Daw Yee Yee Mon	2	1	2	3	3	2	1	-	-	-	2	2..02
5	U Myo Khaing Win	5	2	1	3	2	1	2	-	-	-	1	1.90

4.2. Computing Adjusted Cosine Similarity

Let, U = a set of user (U Myat Htun Kyaw, …, Daw Ei Mon Thu)
u= U Myat Htun Kyaw
i= Candycrushsodasaga
j= Merge Plane
By using equation 3.2,

$$sim(Candycrushsodasaga, Mergeplane) = \frac{\sum_{u\in U}(R_{UMyatHtunKyaw,candycrushsodasaga} - \overline{R}_{UMyatHtunKyaw})(R_{UMyatHtunKyaw,Mergeplane} - \overline{R}_{UMyatHtunKyaw})}{\sqrt{\sum_{u\in U}(R_{UMyatHtunKyaw,candycrushsodasaga} - \overline{R}_{UMyatHtunKyaw})^2}\sqrt{\sum_{u\in U}(R_{UMyatHtunKyaw,Mergeplane} - \overline{R}_{UMyatHtunKyaw})^2}}$$

$$sim(Candycrushsodasaga, Mergeplane) = \frac{\sum_{u\in U}(5-2.24)(4-2.24),(5-2.52)(4-2.52),...,(2-2.05)(1-2.05)}{\sqrt{\sum_{u\in U}(5-2.24)^2(5-2.52)^2...(2-2.05)^2}\sqrt{\sum_{u\in U}(4-2.24)^2(4-2.52)^2...(1-2.05)^2}}$$

sim(Candycrushsodasaga, Mergeplane)= **0.75**

Table 4.2 Calculate Similarity Matrix

	Candy Crush Soda Saga	Merge Plane	Angry Bird	Air Photo Editor College Filter	Block Puzzle	Block Puzzle Conquer	Blossom Blast Saga	-	-	-	Jewels Jungle Match3 Puzzle
Candy Crush Soda Saga	1	0.75	0.11	0.83	0.35	0.23	0.39	-	-	-	0.04
Merge Plane	0.75	1	0.62	0.57	0.04	0.45	0.39	-	-	-	0.99
Angry Birds	0.11	0.62	1	0.75	0.42	0.56	0.97	-	-	-	0.89
Air Camera Photo Editor College Filter	0.83	0.57	0.75	1	0.47	0.64	0.38	-	-	-	0.51
Block Puzzle	0.35	0.04	0.42	0.47	1	0.67	0.70	-	-	-	0.98

4.3. Normalized Rating

The normalized rating calculate to get de-normalize value that is help to find prediction value.

$$NR_{u,N} = \frac{2(R_{u,N} - Min_R) - (Max_R - Min_R)}{(Max_R - Min_R)} \tag{4.2}$$

Where, $R_{u,N}$ is the current rating user u gave item N
$NR_{u,N}$ is the normalized rating

Let, Max_R be the maximum rating =5
Min_R be the minimum rating = 1

e.g

$$NR(UMinMin, Candycrushsodagsaga) = \frac{2(3-1)-(5-1)}{(5-1)} = 0$$

Table 4.3 Calculate Normalization Values for Prediction

Name	Candy Crush Soda Saga	Merge Plane	Angry Birds	Air Camera Photo Editor College Filter	Block Puzzle	Block Puzzle Conquer	Blossom Blast	.	.	.	Jewel Jungle Match
U Min Min	0	-	-0.5	-1	-0.5	0	-1	-	-	-	0
Daw Ei Mon Thu	1	0	0.5	-0.5	-0.5	-1	-	-	-	-	0.5

4.4. Rating Prediction by Using Weighted Sum

Prediction computation method is concerned about predicting the rating for an item to which the user had not rated. The proposed system uses weighted sum method for predicted ratings. Now that we have this nice matrix of similarity values, it would be dreamy if we could use it to make predictions means we are going to predict the rating user u will give item i. This method computes the prediction on an item i for a user u by computing the sum of the ratings given by the user on the items similar to i. Each rating is weighted by the corresponding similarity $s_{i,j}$ between items i and j. The formula for the weighted sum, the prediction $P_{u,i}$ is described below:

$$P_{u,i} = \frac{\sum_{N \in similarTo\,(i)} (s_{i,N} * NR_{u,N})}{\sum_{N \in similarTo\,(i)} (|\, s_{i,N} \,|)} \tag{4.3}$$

Where, $P_{u,i}$ denotes Predicted rating for item i by users u,
$S_{i,N}$ is the similarity between item i and N items (from the similarity matrix)
$NR_{u,N}$ is the normalized rating

Once the ratings are predicted, depending upon the ratings preferences the respected items will be sorted and will be recommended to the user. [5]

e.g

$$P_{UMinMin,Candycrushsodasaga} = \frac{\sum N \in \text{similarTo } i(\text{Candaycrushsodasaga})(|S_{\text{Candycrushsodasaga},N|})}{\sum N \in \text{similarTo } i(\text{Candaycrushsodasaga})(S_{\text{Candyctushsodasaga},N})}$$

$$= \frac{(0.75*0) + (0.11*0.5) + ... + (0.04*0)}{0.75 + 0.11 + ... + 0.04}$$

$= -0.58$

4.5. De-normalization

Finally, the system predicts how will the current user rate application2 and application7. The results are calculated by using equation (3.4) is de-normalization into rating format in the range (1 to 5)

$$R_{u,N} = \frac{1}{2}((NR_{u,N} + 1) * (Max_R - Min_R)) + Min_R \tag{4.4}$$

$R_{u,N}$ is the current rating user *u* gave item *N*

$NR_{u,N}$ is the normalized rating for predict rating value

Let, Max_R be the maximum rating = 5

Min_R be the minimum rating = 1

e.g,

$$R_{u,N} = \frac{1}{2}((-0.58+1)*(5-1)) + 1$$

$$R_{u,N} = \frac{1}{2}(3.68)$$

$$R_{u,N} = 1.84$$

$$R_{u,N} = 2$$

Table 4.4 Table with Prediction Value

No	Name	Candy Crush Soda Saga	Merge Plane	Angry Birds	Air Camera Photo Editor College Filter	Block Puzzle	Block Puzzle Conquer	Blossom Blast Saga	-	-	-	Jewels Jungle Match3 Puzzle	Average
		1	**2**	**3**	**4**	**5**	**6**	**7**	**-**	**-**	**-**	**42**	
1	U Myat Htun Kyaw	5	4	3	2	1	5	3	-	-	-	1	2.24
2	U Min Min	3	2	2	1	2	3	1	-	-	-	2	1.74
3	Daw Nwe Ni Win	5	4	3	2	1	2	2	-	-	-	3	2.52
4	Daw Yee Yee Mon	2	1	2	3	3	2	1	-	-	-	2	2..02
5	Daw Ei Mon Thu	2	1	2	3	3	2	2	-	-	-	2	2.00

5. PERFORMANCE EVALUATION

An evaluation of the recommender system approaches is made to measure the levels of their performances. In this paper, Precision, Recalls are used to evaluate the performance of item-based recommender approach. Precision measures the ability of the system to present only those items that are relevant, and it can be seen as the measure of exactness. Precision measures how many applications retrieved are actually relevant, that is how much result set is on target. For example, 75% precision rate means that 75% of the applications retrieved are relevant, while 25% percent of those applications are not relevant.

5.1. Precision of the System

Precision is defined as the ratio of the retrieved applications that are relevant and the number of all retrieved applications Recommendation is viewed as information retrieval task: Retrieve (recommend) all items which are predicted to be "relevant". Precision is a measure of exactness, determines the fraction of relevant items retrieved out of all items retrieved.
e.g, the proportion of recommended applications that are actually relevant.

$$\text{Precision} = \frac{\text{relevant applications recommended}}{\text{all recommended}} \quad (5.1)$$

$$\text{Precision} = \frac{17}{21} = 0.8095 * 100 = 80.95\ \%$$

5.2. Recall of the System

Recall measures the ability of the system and it can be seen as the measure of completeness. Recall measures how many relevant applications in a collection have actually been found. Recall is defined as the ratio of the relevant recommended and all relevant applications. A measure of completeness, determines the fraction of relevant items retrieved out of all relevant items.
e.g. the proportion of all relevant applications recommended

$$\text{Recall} = \frac{\text{relevant applications recommended}}{\text{all relevant applications}} \quad (5.2)$$

$$\text{Recall} = \mathbf{17/17} = 1 * 100 = 100\ \%$$

5.3. Experimental Results

The objective of this experiment is to verify the rating values which are given by the user that can improve the recommendation accuracy using the collaborative filtering. Precision and Recall are calculated depending on 42 applications with respect to 20 users in table. So, precision average is 81.31 and recall average is 100 as shown in table (5.1).

Table 5.1 Precision and Recall

No	Application Name	Relevant App	All Recommend	All Relevant	Precision	Recall
1	Candy Crush Soda Saga	17	21	17	80.95	100
2	Merge Plane	17	23	17	73.91	100
3	Angry Birds	20	24	20	83.33	100
4	Air Camera Photo Editor Collage Filter	15	21	15	71.43	100
5	Block Puzzle	14	18	14	77.78	100
…	…	…	…	…	…	…
	Average				**81.31**	**100**

6. CONCLUSTION AND FUTURE WORK

Recommendation systems are a powerful tool for extracting additional value for business from its user databases. These systems help users to find the items that the user wants to download or would like to view. This system benefits the users by helping them to find the items they like. New technologies are needed that can dramatically improve the scalability of recommender systems. This system can effectively apply the rating styles in computing similarity for improving the prediction. The demographic similarity of the item can be computed in advance before the prediction computation. They are increasingly used as a tool for e-commerce on the web. This system contributes to provide better quality in predictions than user-based recommendation system. The improvement in quality is consistent over different neighborhood sizes. Moreover, the item neighborhood is fairly static which can be potentially pre-computed and it is possible to retain only a small subset of items and produce reasonably good prediction quality. This system can allow real-time recommendations independent of the size of the user-item matrix. Recommender systems offer users some items that they may desire to get from a web. This system helps user items that they would like to get from web. The advantages of the system are that various ratings on similar items can be accepted. In addition to this, item-based similarity is static. Moreover, time and effort are saving. This recommendation system is not to be end. In the future, this system can be extended to the semantic framework which can apply on mobile computing.

7. REFERENCES

[1] Breese, J.S., Heckerman, D., Kadie, C. "Empirical Analysis of Predictive Algorithms for Collaborative Filtering", In Proc. of the 14th Conf. on Uncertainty in Artificial Intelligence (1998) 43–52
[2] Sarwar, B., Karypis, G., Konstan, J., Reidl, J., "Item-based Collaborative Filtering Recommendation Algorithms", In Proc. of the 10th Int. Conf. on World Wide Web (2001) 50 H.-N. Kim, A.-T. Ji, and G.-S. Jo
[3] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P. Riedl, J, "GroupLens: an open architecture for collaborative filtering of netnews", In Proc. of the ACM Conf. on Computer supported Cooperative Work (1994) 175–186
[4] Lemire, D., Maclachlan, A., " Slope One Predictors for Online Rating-Based Collaborative Filtering", In Proc. of SIAM Data Mining (2005)
[5] Deshpande, M., Karypis, G., "Item-based top-N recommendation algorithms. ACM Transactions on Information Systems", Vol. 22 (2004) 143–177
[6] Zaw, L. H., "Online Car Recommendation Using Collaborative Filtering-Based Agoregated Query (CFAg Query)", May,2016.

[7] Jennifer, P., "Neighborhood-Based Collaborative Filtering", 2016, Switzerland.
[8] Hanafi, Nanna, S., Abdul, S. B., Hasan B., "Collaborative Filtering Implementation In Recommender System", Vol.95, No.16, 31 Aug 2017, Malaysia.
[9] "Implicit ratings and item based filtering",China.
[10] Arissa, A. N., "Recommendation Systems for Mobile Phone Device Application Downloading Using Item Collaborative Filtering Method Implementation", 2009.
[11] Htin, L. A., "Personalized Book Recommendation System By Usig Nrighbourhood-Based Method", 2012.
[12] Aye, A. L., "Comparing Item-Based to User-Based Approach of Collaborative Filtering Methods", March, 2017.
[13] Rohan Katariya, V Krishna Mohan, Shubham Suryawanshi, Ruturaj Dhaneshwar, Geetha Chillarge, "Implementing Collaborative Filtering Approach in Online Event Recommendation System", Vol.6, Issue 5, May 2018, India