MOBILE APP RECOMMENDATION SYSTEM USING K-MEANS AND ITEM-BASED COLLABORATIVE FILTERING

KHIN AYE SAN

M.C.Sc. **DECEMBER 2022**

MOBILE APP RECOMMENDATION SYSTEM USING K-MEANS AND ITEM-BASED COLLABORATIVE FILTERING

BY

KHIN AYE SAN

B.C.Sc.

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Computer Science

(M.C.Sc.)

University of Computer Studies, Yangon

December, 2022

ACKNOWLEDGEMENTS

I would like to express my gratitude and appreciation to the following people who have contributed for completion of this thesis and helped to makes this thesis possible.

Firstly, I am deeply thankful to **Dr. Mie Mie Khin,** Rector of the University of Computer Studies, Yangon, for her continuous and kind guidance during the period of doing this thesis.

My thanks and respects go to **Dr. Si Si Mar Win** and **Dr. Tin Zar Thaw,** Professor, Faculty of Computer Science , University of Computer Studies, Yangon, for her close supervision, invaluable suggestions, kind guidance and constant encouragement during the courseof this thesis.

I am remarkably and especially thankful to my supervisor, **Dr. Thu Thu Zan**, Associated Professor, Faculty of Computer Science, University of Computer Studies (Sittway), for her close supervision, invaluable suggestions, kind guidance and constant encouragement duringthe course of this thesis.

I also wish to express my gratitude to **Daw Mya Hnin Mon**, Associated Professor, Department of Languages, University of Computer Studies, Yangon, for editing of my thesis from the language point of view.

I am also grateful to express my gratitude to all teachers from the University of Computer Studies, Yangon for teaching and guiding us throughout the period of study. I would also like to extend my thanks to my beloved parents for their support not only for the fulfillment of the degree of Master of Computer Science but also my life.

Finally, I would like to thank all of the teachers and staffs from University of Computer Studies, Yangon and Sittway for their support. I also thank my friends and colleagues for supporting me in various ways to do this thesis.

ABSTRACT

Recommender Systems are being widely used in many application settings to suggest products, services, and items to potential users. They are the software techniques providing suggestions for items to be of use to a user. The main purpose of Recommender Systems is to generate meaningful recommendations about the items to a collection of users for their interested items. A variety of approaches in recommendation are user- based collaborative filtering, item-based collaborative filtering, model- based collaborative filtering, content-based recommendation, context- aware recommendations and so on. However, there are two main approaches in recommendation: user-based and item-based collaborative filtering and the difference between them is that user-based takes the users' behavior and item-based takes items' rating values for similarity measurement. Since the computational complexity of user-based recommendation grows linearly with the number of users, item-based recommendation techniques have been developed. The goal of this system is to provide meaningful recommended applications to the mobile phone users that are relative to their needs or targets. This system uses K- meansclustering algorithm to cluster the users based on their age and rating values and item-based collaborative filtering method based on rating values of theitems. By using this system, the mobile phone users can get very effectiverecommendations about applications without waste of time and effort.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

CHAPTER 1
INTRODUCTION

In recent years, the World Wide Web (WWW) has emerged as an important and dynamic source of information. The researchers from many areas of natural language processing (NLP), artificial intelligence (AI), information extraction, and ontology building for the semantic web are attracted by WWW.

Because of the improvement of internet, many e-commerce jobs intend to develop recommender systems. The purpose of recommender system to identify and support the set of items to users. In this way, many commercial recommendation systems consider to use extensively user-based collaborative filtering. However, typical commercial applications can cost several millions because the computational complexity of this method grows linearly with the number of customers. The item-based recommendation technique has been developed to address these scalability concerns. This technique mainly uses the user-item matrix to find the relations between items and the relations to compute the recommendation list.

As the volume of information available to internet users continue to grow at unprecedented rates, systems that are able to filter out and present relevant information in a meaningful way become more and more important. Such systems are referred to as recommender systems. The recommender system firstly predicts the rating values of items that are not rated. Then, the system would assign the rating values of items and recommend the items with highest ratings.

Recommender systems are very popular for the online users who want to know which are up-to-date and suitable for them. Therefore, many online systems target to update their systems using these techniques and recommend applications based on the item-based collaborative filtering. However, users cannot satisfy their recommendation lists. Some users need to know more than their provided information. Many systems let the users to write what they need to know and the different users have different requirements. They reveal their requirements with different forms of writing. Most systems cannot retrieve what the users really want.

In this approach, many websites were used to handle the user inputs by using information retrieval system. But this can only provide to the users the matching information relevant with users' input. These systems may have limitations because the users have a few chances to their desired items. Therefore, the ecommerce sites should develop to allow the users to enter free text as they desire without worrying about the structures and grammars. The researches tried to build the autonomous system which recognizes the user's desires.

Then, the recommendation system has been a recent focus of researchers and practitioners. It filters the information and applies data analysis and helps the user in products finding. Therefore, the users or customers can search easily products that they want to purchase. The system will also recommend the products that the customers may purchase. It provides a personalized solution with automated and sophisticated decision support system by reducing complicated searching process. Most e-commerce sites use recommender systems to recommend products and help users in product purchasing.

The recommender system allows e-commerce websites to suggest products to their customers by providing relevant information to assist them in shopping tasks. By selecting a subset of items from a universal set based on user preferences, recommendation systems attempt to reduce information overload and retain user.

In most recommender systems, recommendations need to be calculated in real time, meaning functions used in the system must run quickly on large datasets while still returning accurate predictions. The system must also handle what is known as the cold-start problem. When a user or item is first added to the system, very little is known of them, making comparisons to other users or items already in these systems very difficult. Any given recommender system consists of various elements, and each element must be designed with these goals in mind.

Typical recommender systems use a recommendation task with a static view but treat prediction problem. Therefore, they apply prediction algorithms to provide items that relative with their interested items.

Internet provides the information and resources to online users so that they can confuse between information about items and cannot decide which one would like to buy over the Internet This system describes and presents the most similar results between items over a proposed rating matrix to choose the items.

The collaborative filtering is becoming very popular to reduce information conflicts. The main task of collaborative filtering is to create a database of user preferences and items. Item-based method intends to avoid the bottleneck by determining relationships between items rather than users. Recommendation results are provided by computing similarity between items the user has liked. Item-based methods provide same quality results because relationships of the items are static.

Recommendation system is being used extensively in many domains such as movies, news and books because it is such a kind of information filtering. This system considers item-based collaborative filtering method although there are different types of approaches in developing recommender system. The item-based collaborative consists of cosine-based similarity, model-based similarity, memory-based similarity, correlation-based similarity, adjusted cosine similarity, coefficient similarity.

With a rapid increase of mobile applications available for downloading, it has become a very difficult task for the users to find exactly what they want. In order to choose one among all of the offered applications, they have to put in a lot of effort and waste a lot of time. To make things easier for the users, android application markets are using recommender systems which provide users with application suggestions.

The purpose of this system is to recommend the mobile applications for the users that are relative to their wanted applications. In this system, the K-means clustering algorithm is used to cluster the registered users based on their age and rating values of applications. One such class of item-based recommendation algorithm is applied to determine similarities between items and uses them to identify the recommended items. The main steps of this system are as follows:

(i) Clustering algorithm is used to cluster the registered users,
(ii) Computing the similarity values to find and recommend similar applications, and
(iii) Recommendation results are displayed with computed similarity measures between similar items with users' interested applications

1.1 Objectives of the Thesis

The objectives of the thesis are:

- ❖ To provide the relevant mobile applications for the users' interests as the recommended results.
- ❖ To save time and effort in searching and downloading of the interested mobile phone applications for the users.
- ❖ To enhance the accurate recommendations by using item-based collaborative filtering method and k-means
- ❖ To understand the item-based collaborative filtering

recommender system

1.2 Organization of the Thesis

The goal of this system is to recommend mobile applications by using item-based collaborative filtering for the users. The body of the thesis consists of five chapters.

Due to the improvement of world wide web, the importance of recommender system is introduced in chapter 1. In this chapter, the different types of collaborative filtering techniques are also briefly described.

The theoretical background of the recommender system and the different collaborative filtering methods are explained in chapter 2. In chapter 3, it describes the steps of the proposed item-based collaborative filtering system such as the clustering algorithm, collaborative filtering method and similarity computation.

Then, chapter 4, it describes the implementation of the system. Moreover, this chapter discusses about the dataset, detailed calculation formula, and the results. And then chapter 5 as the final chapter describes conclusion and directions for future work.

CHAPTER 2

THEORETICAL BACKGROUND

2.1 World Wide Web

The World Wide Web (WWW) [24] is a system of inter linked hypertext web documents that can be accessed via the Internet. The user can view web pages including text, images, videos and other multimedia and navigates between them with hyperlinks using web browser [12]. World Wide Web is a collection of globally distributed text and multimedia documents and files. The Web uses hypertext technology to represent an interface for users and can be accessed to retrieve specially formatted documents.

To define the structure of web pages, Hypertext Markup Language (HTML) [28] is used because it contains embedded images, texts, links to other pages and programming codes. By using the hyperlinks, the users can navigate to other referenced web pages. To facilitate data sharing over the internet, another markup language, XML, has been defined.

- The Hypertext Transfer Protocol (HTTP) [27] can be defined as a protocol to transmit web pages over world wide web. It specifies how servers and clients communicate with web page URL. By using web browser, the users perform http request to server. Then, the server responses web page content.

- The Uniform Resource Locator (URL) is a universal system for referencing resources on the Web. It is standardized naming convention for websites or webpages and also a universal system to refer the resources, webpages or files.

These techniques form together an information sharing platform. WWW has become the explosive growth and internet access is available everywhere. Due to of this fact, web pages complexity and size have grown. Therefore, the research works have been developed many systems to help users in seeking useful information using filtering approach [4].

2.2 Related Work

A Recommender System (RS) [19] can be defined as web-based system to provide suggestion about the items to users that want to search. A recommender system also supports personalization individually for each customer by recommending items as customization. The steps are (1) acquisition of users' preferences; (2) computation of recommendation results using appropriate results; and (3) presentation of the computed results.

The main purpose is to provide useful recommendations to users and display similar items that are related interested items. The real-world examples are movies on Netflix and book suggestions on Amazon. The design engines will be different depending on domains and data characteristics available. As an example, Netflix uses the rating scale from 1 to 5 so that movie watchers give rating scale form liked to disliked (5 to 1). The relationship quality between items and users are recorded by the system. Recommender systems differ in analyzing methods of data sources.

While content-based filtering methods are based on profile attributes and hybrid techniques attempt to combine both of these designs, collaborative Filtering systems analyze historical interactions alone. The real-world problems are facing as an active research area about the recommendation architecture [11].

There has been many researches and approaches about recommender systems by using collaborative filtering. In the study [13], the merging of content and user-item based collaborative filtering approach was proposed that generates relatively small number of recommendations. This study contributes to help better items with fewer recommendation list to curtail recommendation size.

The research work [14] proposed an algorithm to balance three current similarity measurements such as: Adjusted cosine similarity, Cosine-based similarity, and Pearson correlation similarity. In this study, there is a comparison between the improved algorithm of traditional measurement metrics and the existing algorithm of the traditional metrics.

A novel recommender system that assist mobile game users in recommending useful and effective mobile game applications was proposed in the study [12]. This study emphasizes only on item-based collaborative filtering method to recommend game applications.

The researchers in the study [15] proposed a hybrid model to achieve high-quality e-commerce recommendations. The proposed model based on the effective combination of collaborative filtering techniques. The model consists of the following components: item-based collaborative filtering and user-based collaborative filtering. This model also computed the similarity values between the predicted objects. To introduce the recommender system for the medium-scale e-commerce platforms this study can become a methodological basis.

The research in the study [16] presented a modest approach to enhance prediction in Movie Lens dataset with high scalability by applying user-based collaborative filtering methods on clustered data. The research work [17] presented a novel approach for item-based collaborative filtering, by leveraging BERT to understand items, and score relevancy between different items. This method could address problems that plague traditional recommender systems such as cold start, and "more of the same" recommended content.

The internet [31], the main largest source of information with hundreds of millions of pages worldwide. Information retrieval systems are the most useful tools to guide users' information searching. The users want personalized search systems, without limiting to relevant items retrieval. This is main goal of recommender systems. They use information about users, user profiles, to predict the utility or relevance of a particular item, thus providing personalized recommendations.

Until now, recommender systems have been used basically in two tasks. First, they have been used to predict the utility of a given item to the user. In this task, often known as annotation in context, the user first selects the item (or items) in which he/she is interested. The recommender system then predicts the rating the user would give to that item. Second, recommender systems have been used, to recommend a list of items to the user. In this case, often called the find good items task, the system chooses the items that it considers the most relevant. Actually, recommender systems can also be used for other tasks, such as find all good items recommended sequence, just browsing or find credible recommender, although these have not yet attracted much interest among researchers [2].

The recommender systems [20] employ various algorithmic strategies that provide suggestions, such as what application to download and what game to play for items related to various user-specific decision-making processes. They are also intended to help individuals filter the potentially overwhelming information available on relevant websites. In their simplest form, recommendations are presented as personalized ranked lists of items which are predictions of games or applications that best match a user's preferences and constraints. The matching process must construct a database of user preferences from an array of possible sources: preferences explicitly expressed as in filling out a survey specifying ratings for products, implicitly inferred by interpreting user actions such as past purchases (it is common practice, for example, to treat "click troughs (CTs)" to a particular web page as a proxy for user preferences for the product(s) advertised on that page), and/or gathered from 3rd party (primarily demographic) data sources. Additionally, the system may have access to item-specific profile attributes such as product descriptions [18].

Recommender systems can be classified the data sources manipulation and potential match identification between users and items. The technique can be slightly different but primarily focus on collaborative filtering, content-based, community-based, knowledge-based and hybrid [6]. They adapt to give suggestions to users based on preferences by providing personalization on ecommerce sites. So, many commercial sites provide in choosing many products that meet with consumers' needs.

Amazon [25], and Netflix [26], largest ecommerce sites are using recommender systems in choosing products to buy and rent. These sites provide advices to users based on previously rated items. Users receive recommendation list based on the applications that have already rated. The recommendation list is displayed that are similar interested applications and hence, users can explore items that are unaware previously.

2.3 Collaborative Filtering Recommender System

The most successful recommendation technology is collaborative filtering recommender system [21]. This approach is automatic that evaluates target items based on users' opinions. Collaborative filtering recommender system can be described as people-to-people correlation as they recommend products to a potential user based on the degree of correlation between that user and other users who have purchased the products in the past. This approach assumes that human preferences are correlated, in that a user with similar asses will rate things similarly. Thus, the typical input of this approach are explicit ratings.

Collaborative knowledge sources are relied in collaborative filtering approach. Thus, any other information about users or items are not required in this approach. Many domains other than text-based items applied them in content-based. The various domains such as Group lens, Amazon implement collaborative filtering systems [11].

Collaborative filtering do the filtering activities by focusing on similarity of customer characteristics and items attributes to give new information to customers. It processes data filtering approach and evaluate items with active user evaluation to get rating values. The system filters item list to give information based on likeness patterns. The group members' interest can be classified with new beneficial category for other members.

There are three steps in recommendation process: searching similar user, making neighborhood, and counting prediction based on selected neighbors. Different rating styles are (1) numbering styles, (2) good or bad (3) unary value.

To find similarity in two forms using collaborative filtering. *Prediction-* numerical value that do not give rating vales.

Recommendation- user like most item recommendation list. There are two main approaches: user-based and item-based [3].

Some recommender systems use collaborative filtering and it has two senses, a narrow one and a more general one. Their applications typically involve very large datasets. Collaborative filtering methods have been applied to many different kinds of data including sensing and monitoring data, such as in mineral exploration, environmental sensing over large areas or multiple sensors financial data, such as financial service institutions that integrate many financial sources or in electronic commerce and web applications where the focus is on user data, etc. Although some of the methods and approaches may apply to the other major applications, the remainder focuses on collaborative filtering for user data,. Collaborative filtering methods have user-based nearest neighbor algorithms and item-based nearest neighbor algorithms [8].

Collaborative filtering is probably the most widely implemented and best understood technique. The collaborative filtering is a process of filtering or evaluating items using the opinions of others. The collaborative filtering is to filter data based on the similarity of the characteristics of the consumer so it can provide new information to the consumer group that is almost the same. The difference in interest in some members of the group creates new source information that may be useful to other group members.

In general, the process of recommendation consists of three steps: the similar user discovering, make the neighborhood, and the prediction calculation based on a selected neighbor. Collaborative filtering generates predictions or recommendations to users or subscribers intended for one or more items. Items can consist of anything that can be provided by someone such as books, movies, art, article, or tourisms destination.

Unavailability rating indicates there is no information linking users with an item. The rating values can be aquired explicitly, implicitly, or a combination of both. Explicit rating that is obtained opinions on certain items. Implicit rating that is obtained through the action of the customer [1].

There are many recommendations techniques such as the following;

(1) User-based Collaborative Filtering

(2) Item-based Collaborative Filtering

(3) Memory-based Collaborative Filtering

(4) Model-based Collaborative Filtering and

(5) Content-based Collaborative Filtering

2.3.1 User-based Collaborative Filtering Method

The purpose of user-based to compute similarity between users preferences and it compute the similarity values and give recommendations. When there is large number of forms, this system may sometimes perform inefficiently [5]. The user's preferences are taken and find a set of users, similar users with target user.

After a set of neighbors has been formed, different algorithms are used to combine favorite neighbors to generate predictions or recommendations for the users. User-based nearest neighbor algorithm [22] used statistical methods to find a set of neighbors based on unique weighting values. These values must be historical agreement with target user. The system uses different procedures to merge the preferences of neighbors and produces N-top predicted results or recommended item group for active users once neighbor group is formed. Then, it provides facilities to users by giving highest purchase value. This method has a solution for limitation problem, scalability and memory, time issues.

User-based nearest neighbor algorithm [22] uses statistical techniques to find a set of users, known as neighbors, that have history agrees with the target user. After a set of neighbors is formed, the system uses different algorithms to combine favorite neighbors to generate predictions or recommendations for the active N-top users.

The Collaborative Filtering (CF) is a user- based, or user-user, form, which recommends to an individual user, items that other users with similar tastes liked in the past. The Collaborative Filtering (CF) is essentially item-agnostic, focusing instead upon users' ratings of items rather than attributes of the items themselves. The main challenge in implementing Collaborative Filtering (CF) is to define effective measures of similarity based upon the comparative ratings history of the users under consideration [1].

2.3.2 Item-based Collaborative Filtering Method

Item-based collaborative filtering mainly uses rating values of items that are given by users. The method needs to get the users usability for making recommendations. Thie method can solve the problems of user-based such as scalability and limitations of memory and time) [1].

The difference between scalability and locality provides an efficient way for maintaining the partition of user structure [4]. 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 $\{s_{i1}, s_{i2}, \ldots\ldots, s_{ij}\}$ are also computed [9]. The different descriptions of user-based and item-based collaborative filtering are presented in Figure 2.1 and 2.2.

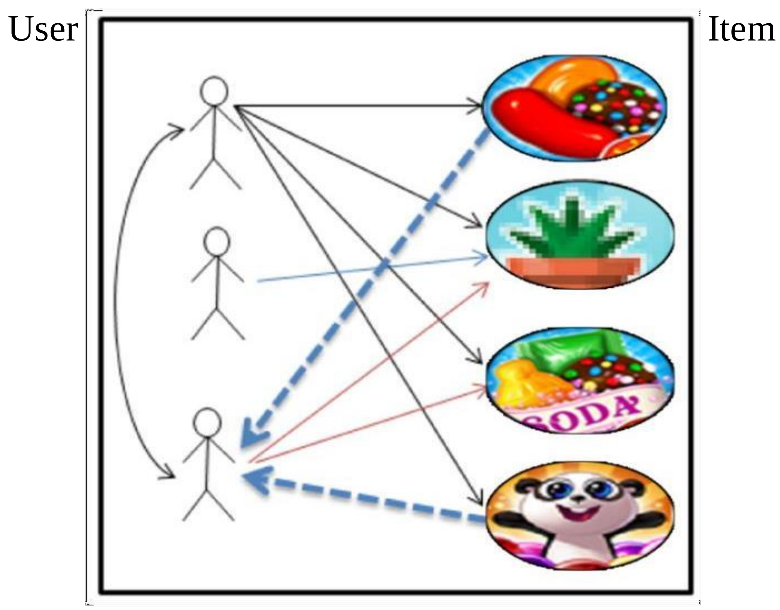

Figure 2.1. User-based Collaborative Filtering

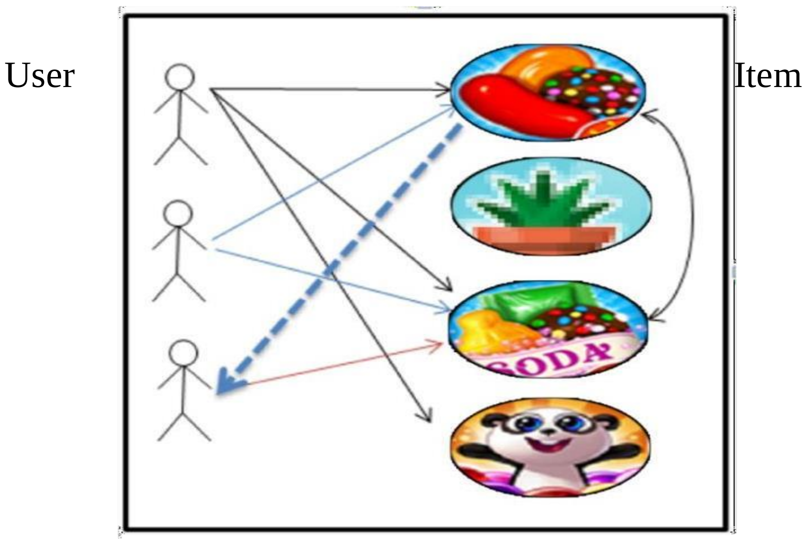

Figure 2.2. Item-based Collaborative Filtering

2.3.3 Memory-based Collaborative Filtering Method

Memory-based algorithms [29] utilize the entire user-item database to generate a prediction. These systems employ statistical techniques to find a set of users, known as neighbors, that have a history of agreeing with the target user (i.e., they either rate different items similarly or they tend to buy similar set of items). Once a neighborhood of users is formed, these systems use different algorithms to combine the preferences of neighbors to produce a prediction or top-N recommendation for the active user. The techniques, also known as nearest-neighbor or user-based collaborative filtering, are more popular and widely used in practice.

The memory-based algorithms can be stored in memory. The recommendation results are calculated based on the entire user's database. The recommendations process for the memory-based algorithm includes user profiling, neighborhood formation and recommendation generation. It first builds an interest profile for a user based on the user's ratings on items that the user has purchased before, and then it makes recommendations based on the similarity between the interest profile of that user and those of the other users. Thus, searching for similar preferences between the active user and the other users is an important step in this recommendation approach before presenting the recommendation according to the preference of similar users [11].

2.3.4 Model-based Collaborative Filtering Method

Model-based collaborative filtering algorithms provide item recommendation by rest developing a model of user ratings. Algorithms in this category take a probabilistic approach and envision the collaborative filtering process as computing the expected value of a user prediction, given his/her ratings on other items. The model building process is performed by different machine learning algorithms such as clustering, and rule-based approaches. Clustering model treats collaborative filtering as a classification problem and works by clustering similar users in same class and estimating the probability that a particular user is in a particular class C, and from there computes the conditional probability of ratings. The item recommendation based on the strength of the association between items

[10].

The model-based approaches use these ratings to learn a predictive model. The general idea is to model the user-item interactions with factors representing latent characteristics of the users and items in the system like the preference class of users and the category class of items. This model is then trained using the available data, and later used to predict ratings of users for new items. [2]

The model-based algorithm is learned from the collection of ratings based-on the training data. Then, the model validity is checked with the testing data and finally the rating predictions of the target user's no-rating products are computed.

The model-based methods are not always as fast and scalable as we would like them to be, especially in the context of actual systems that generate real-time recommendations on the basis of very large datasets. To achieve these goals, model-based recommender systems are used. The model –based methods involve building a model based on the dataset of ratings. Model-based methods extract some information from the dataset, and use that as a model to make recommendations without having to use the complete dataset every time. This approach offers the benefits of both speed and scalability [11].

2.3.5 Content-based Collaborative Filtering Method

This approach gives recommendations based on rating values of target user to a particular item and rely on personalized recommendation generally. When a particular item has been rated by customer, this method is suitable to use in recommender system. The major role is previous user experiences in this technique.

The content-based filtering methods recommend items similar to those a given user has liked in the past. The content-based methods need proper techniques for representing the items and producing the user profile, and some strategies for comparing the user profile with the item representation [8].

2.4 Similarities in Collaborative Filtering

The different types of similarities in collaborative filtering are as follows:

- Cosine-based Similarity
- Correlation-based Similarity
- Adjusted-cosine Similarity

2.4.1 Cosine-based Similarity

Most recommender systems use cosine based similarity to compute similar items so that it is very popular [11].The cosine-based similarity works on the concept of statistical cosine where two items are considered as two vectors in the dimension m user space. [6] 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

$$\mathrm{sim}(i, j) = \cos(i, j) = . \frac{i.j}{||i||2 * ||j||2} \quad (2.1)$$

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

This formalism can be adopted in collaborative filtering which uses users or items instead of documents and ratings instead of word frequencies [2].

$$sim(i,j) = \frac{\sum_{u \in U}(R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{\sqrt{\sum_{u \in U}(R_{u,i} - \bar{R}_i)^2}\sqrt{\sum_{u \in U}(R_{u,j} - \bar{R}_j)^2}} \qquad (2.2)$$

2.4.2. Correlation-based Similarity

The correlation similarity is does not consider the differences in the mean and variance of the ratings. The correlation has been used widely in statistical term. It uses similarity between two items which measured by calculating correlation of the set of users who rated the set of items. The correlation represents by similarity between item i and j and also the rating values that are owned by each union (U) which composes by u, i and j. the similarity between i and j is given in equation (2.2) [3].

2.4.3. Adjusted Cosine Similarity

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 (2.3) [3].

There are multiple options related to choosing the similarity measure. Pearson correlation [23], cosine vector similarity and adjusted-cosine similarity is some of the well knows similarity measures used to compute the similarity. One fundamental difference between the similarity computation in user-based collaborative filtering and item-based collaborative filtering is that in case of user-based collaborative filtering the similarity is computed along the rows of the matrix but in case of the item-based collaborative filtering the similarity is computed along the columns. The formulation for adjusted-cosine similarity is described below:

$$sim(i,j) = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_u)(R_{u,j} - \bar{R}_u)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_u)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_u)^2}} \quad (2.3)$$

Where $R_{u,i}$ is rating of user u for item i,

$R_{\hat{u}}$ is average rating of user u for all items,
$R_{u,j}$ is rating of user u for item j.
$u \in U$.

One critical step in the item-based collaborative filtering algorithm is to compute the similarity between items and then to select the most similar items. The basic idea in similarity computation between two items i and j is to first isolate the users who have rated both of these items and then to apply a similarity computation technique to determine the similarity $s_{i,j}$. Figure 2.3 illustrates this process, here the matrix rows represent users and the columns represent items. There are a number of different ways to compute the similarity between items. Here three such methods me presented. These are cosine-based similarity, correlation-based similarity and adjusted-cosine similarity.

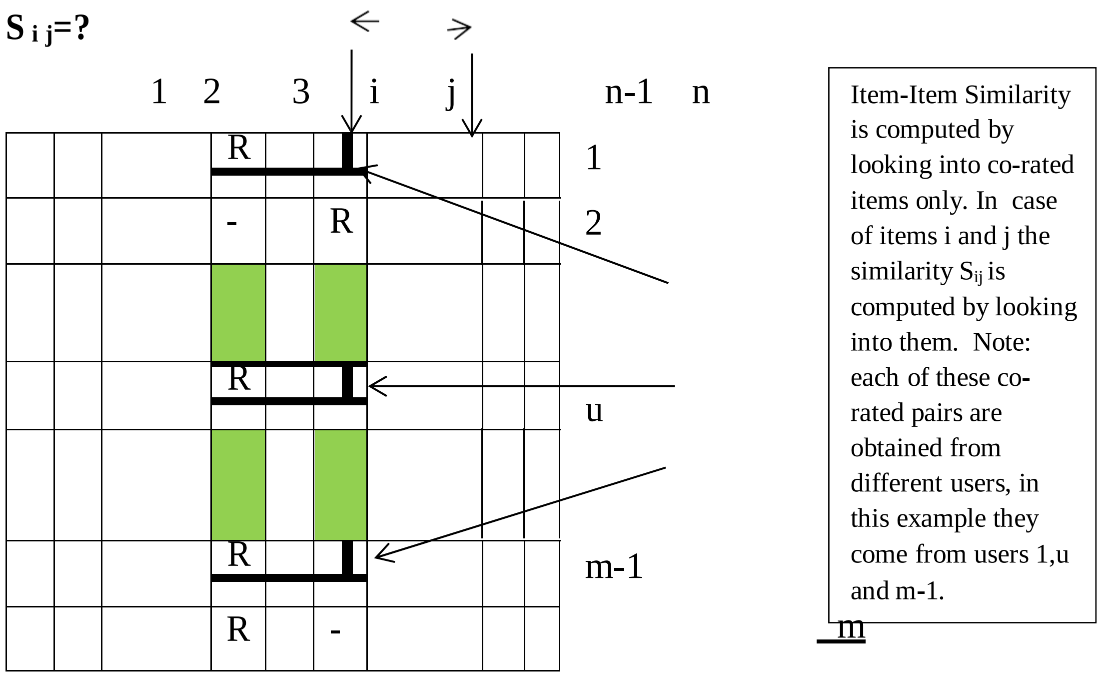

Figure 2.3. Isolation of the Co-rated Items and Similarity Computation

One fundamental difference between the similarity computation in user-based Collaborative Filtering and item-based Collaborative Filtering the similarity is computed along the rows of the matrix but in case of the item-based Collaborative Filtering the similarity is computed along the columns i.e., each pair in the co-rate set corresponds to a different user. Computing similarity using basic cosine measure in item-based case has one important drawback the difference in rating scale between different users are not taken into account.

The adjusted cosine similarity offsets this drawback by subtracting the corresponding user average from each co-rated pair [7].

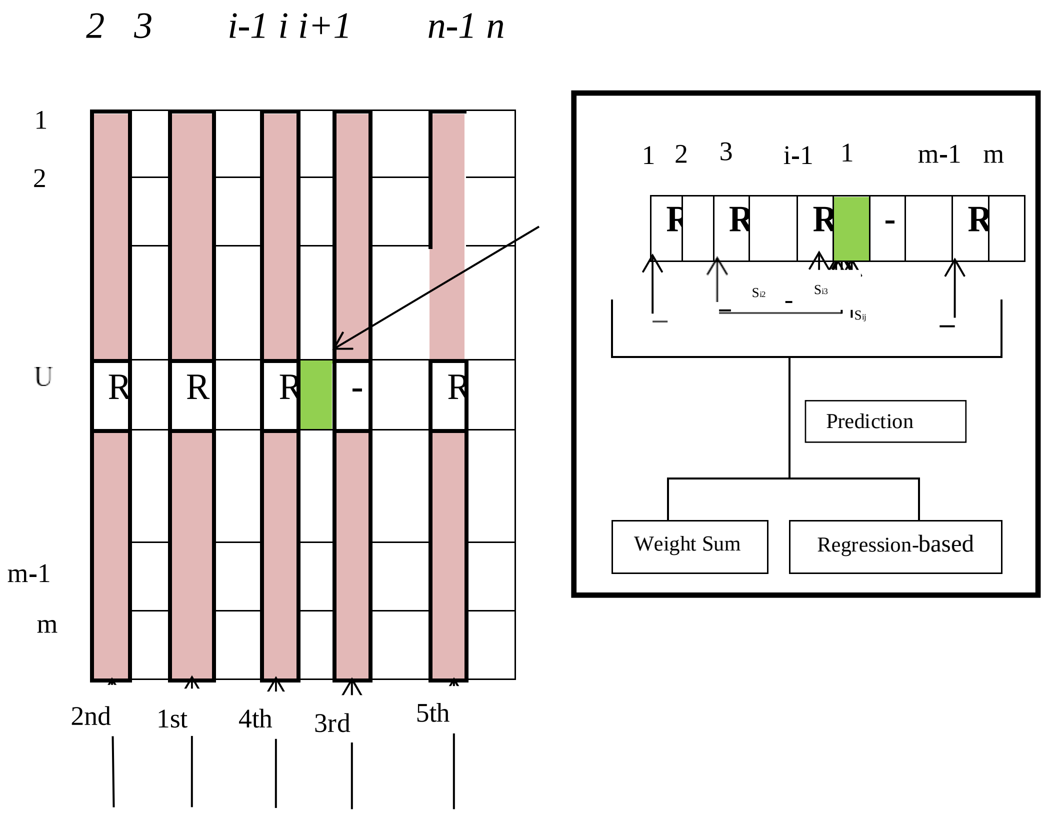

Figure 2.4 Item-based Collaborative Filtering Algorithm.

CHAPTER 3

RECOMMENDATION SYSTEM USING K-MEANS AND ITEM-BASED COLLABORATIVE FILTERING

3.1 Recommendation System

The recommendation systems employ various algorithmic strategies that provide suggestions for items related to various user- specific decision-making processes, such as what application to download and what game to play. The recommendation systems become important tools for internet downloading activities in e-commerce as they can provide a personal item for each user and support the user in product searching. They provide personalization on e-commerce sites by adapting product suggestions according to each user's preferences. The recommender systems also help e-commerce sites achieve mass customization by providing multiple choices of products that meet the multiple needs of multiple consumers. Although there are many recommender systems, this system uses k-means clustering and item- based collaborative filtering.

3.2 K-Means Clustering of the System

K-Means Clustering is an unsupervised learning algorithm that is used to solve the clustering problems in machine learning or data science which groups the unlabeled dataset into different clusters. Here, K defines the number of pre-defined clusters that need to be created in the process, as if K=2, there will be two clusters, and for K=3, there will be three clusters, and so on. It allows us to cluster the data into different groups and a convenient way to discover the categories of groups in the unlabeled dataset on its own without the need for any training. It is a centroid-based algorithm, where each cluster is associated with a centroid.

The main aim of this algorithm is to minimize the sum of distances between the data point and their corresponding clusters. The algorithm takes the unlabeled dataset as input, divides the dataset into k-number of clusters, and repeats the process until it does not find the best clusters. The value of k should be predetermined in this algorithm.

The k-means clustering algorithm mainly performs two tasks:

(1) Determines the best value for K center points or centroids by an iterative process.

(2) Assigns each data point to its closest k-center. Those data points which are near to the particular k-center, create a cluster.

Hence each cluster has datapoints with some commonalities, and it is away from other clusters. The working of the K-Means algorithm is explained in the below steps:

Step-1: Select the number K to decide the number of clusters.

Step-2: Select random K points or centroids. (It can be other from the input dataset).

Step-3: Assign each data point to their closest centroid, which will form the predefined K clusters.

Step-4: Calculate the variance and place a new centroid of each cluster.

Step-5: Repeat the third steps, which means reassign each datapoint to the new closest centroid of each cluster.

Step-6: If any reassignment occurs, then go to step-4 else go to FINISH.

Step-7: The model is ready.

Figure 3.1. shows the users clustering based on their age and average rating values of applications.

Table Rating Users and Items				Cluster Centroids				
No	User Name	Age	Avg Rating	Cluster1 [17,1] k1	Cluster2 [20,2] k2	Cluster3 [19,2] k3	Minimum Distance Value	Cluster Number
				Formula D=sqrt((x2-x1)^2+(y2-y1)^2)				
1	Mg Aung Myat	18	2.24	1.59	2.01	1.03	1.03	3
2	Mg Naing Lin	20	1.74	3.09	0.26	1.03	0.26	2
3	Ma Phoo Pwint Thazin	19	2.52	2.51	1.13	0.52	0.52	3
4	Ma Yoon Nadi Zaw	20	2.02	3.17	0.02	1.00	0.02	2
5	Mg Myo lin	17	1.90	0.90	3.00	2.00	0.90	1
6	Mg Aung Bo Bo Zaw	20	2.45	3.33	0.45	1.10	0.45	2
7	Ma Khin Htet Htet Kyaw	18	2.07	1.47	2.00	1.00	1.00	3
8	Ma Ei Thazin Phyo	17	1.90	0.90	3.00	2.00	0.90	1
9	Ma Khin Myat Thu	19	2.00	2.24	1.00	0.00	0.00	3
10	Ma Su Myat Wai	20	2.24	3.25	0.24	1.03	0.24	2
11	Ma Hla Hla Aye	21	1.79	4.08	1.02	2.01	1.02	2
12	Ma Tin Zar Wine	22	2.52	5.23	2.07	3.05	2.07	2
13	Ma Aye Myat Thu	18	2.02	1.43	2.00	1.00	1.00	3
14	Mg Myo Thant Zaw	17	1.90	0.90	3.00	2.00	0.90	1
15	Ma Khin Myo Thant	19	2.45	2.47	1.10	0.45	0.45	3
16	Mg Kyaw Myint	20	2.07	3.19	0.07	1.00	0.07	2
17	Ma Thuzar Wai	19	2.24	2.35	1.03	0.24	0.24	3
18	Mg Zaw Mn Naing	17	1.79	0.79	3.01	2.01	0.79	1
19	Mg Win Mg Oo	18	2.52	1.82	2.07	1.13	1.13	3
20	Ma Khin Thu zar	21	2.00	4.12	1.00	2.00	1.00	2

Figure 3.1. User Clustering using K-Means

3.2 Item-Based Collaborative Filtering of the System

There are eight calculation steps behind the process of the system as follows:

(1) Average Rating Calculation

(2) Calculate the clusters by users based on age and rating values

(3) Determine target rating list

(4) Computer the similarity values between applications with adjusted cosine similarity.

(5) Build the similarity matrix

(6) Calculate normalization rating values

(7) Compute the predicted rating values using weighted sum

(8) Calculate de-normalization target rating values.

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.

3.3.1 Computing Average Rating

Total rating of Mg Aung Myat divided by number of applications

.

$$\text{Average Rating} = \overline{R}_u = \frac{\text{Total of rating of user}}{\text{Number of application}} \quad (3.1)$$

Let, Average Rating

$$\overline{R}_u = \frac{94}{42} = 2.238$$

$$= 2.24$$

So, Average Rating of Mg Aung Myat is 2.24

Table 3.1 Calculate 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	Mg Aung Myat	5	4	3	2	1	5	3	-	-	-	1	2.24
2	Mg Naing Lin	3	0	2	1	2	3	1	-	-	-	2	1.78
3	Ma Phoo Pwint Thazin	5	4	3	2	1	2	2	-	-	-	3	2.52
4	Ma Yoon Nadi Zaw	2	1	2	3	3	2	1	-	-	-	2	2..02

5	Mg Myo lin	5	2	1	3	2	1	2	-	-	-	1	1.90
6	Mg Aung Bo Bo Zaw	4	3	2	1	2	3	2	-	-	-	3	2.45
7	Ma Khin Htet Htet Kyaw	5	4	3	2	1	2	3	-	-	-	3	2.07
8	Ma Ei Thazin Phyo	5	2	1	3	2	1	3	-	-	-	1	1.90
9	Ma Khin Myat Thu	5	2	1	1	2	1	2	-	-	-	5	2.00
10	Ma Su Myat Wai	5	4	3	2	1	5	3	-	-	-	1	2.24
11	Ma Hla Hla Aye	3	2	2	1	2	3	1	-	-	-	2	1.79
12	Ma Tin Zar Wine	5	4	3	2	1	2	2	-	-	-	3	2.52
13	Ma Aye Myat Thu	2	1	2	3	3	2	1	-	-	-	2	2.02
14	Mg Myo Thant Zaw	5	2	1	3	1	3	2	-	-	-	1	1.90
15	Ma Khin Myo Thant	4	3	2	1	2	3	2	-	-	-	3	2.45
16	Mg Kyaw Myint	5	4	3	2	1	2	3	-	-	-	3	2.07
17	Ma Thuzar Wai	5	4	3	2	1	5	3	-	-	-	1	2.24
18	Mg Zaw Mn Naing	3	2	2	1	2	3	1	-	-	-	2	1.79
19	Mg Win Mg Oo	5	4	3	2	2	2	2	-	-	-	3	2.52
20	Ma Khin Thu zar	2	1	2	3	3	2	0	-	-	-	2	2.05

3.3.2 Adjusted Cosine Similarity

Unlike the user-Based collaborative filtering, the item-based approach looks into the set of items, the target user has rated and computes how similar they are to the target item i and then selects j most similar items ($i_1, i_2, i_3, \ldots, i_j$). At the same time, their corresponding similarities (S_{i1}, S_{i2}, S_{i3}, …, S_{ij}) are also computed (the user rated both i and j) as shown in equation 3.1. Let the set of users who rate both i and j are denoted by U, then the Adjusted Cosine Similarity is given by: equation 3.2.

This method appears as a solution to several problems in user- based collaborative filtering because it has the problem of limited and scalability as well as the problems of time and memory. In this method, the rating value is called the value of similarity among the items that given user. [8]

By equation (3.2), the similarity between items *i* and *j* using

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

Where $R_{u,i}$ is the rating value of item i given by user

$R_{u,j}$ is the rating value of item j given by user

$\bar{R}_u$ is the average of the u^{th} user's ratings

e.g

Let, U = a set of user (Mg Aung Myat, …,Ma Khin Thu Zar)

u=Mg Aung Myat

i= Candycrushsodasaga

j= Merge Plane

By using equation ,

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

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

sim (Candycrushsodasaga, Mergeplane)= 0.75

Table 3.2 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
Block Puzzle Conquer	0.23	0.45	0.56	0.64	0.67	1	0.03	-	-	-	0.58
Blossom Blast Saga	0.39	0.39	0.97	0.38	0.70	0.03	1	-	-	-	0.62
-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-
Jewels Jungle Match 3 Puzzle	0.04	0.99	0.89	0.51	0.98	0.58	0.62	-	-	-	1

3.3.3 Normalized Rating

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

$$NR_{u,N} = \frac{2(R_{u,N} - Min_R) - (Max_R - Min_R)}{(Max_R - Min_R)} \qquad (3.3)$$

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 3.3 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 Saga	-	-	-	Jewel Jungle Match 3Puzzle
Mg Naing Lin	0	-	-0.5	-1	-0.5	0	-1	-	-	-	0
Ma KhinThu Zar	1	0	0.5	-0.5	-0.5	-1	-	-	-	-	0.5

3.3.4 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 the nice matrix of similarity values would be dreamy if it is use to make predictions means that 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:

$$P_{u,i} = \frac{\sum_{N \in similarTo(i)} (s_{i,N} * NR_{u,N})}{\sum_{N \in similarTo(i)} (|s_{i,N}|)} \qquad (3.4)$$

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

$$P_{\text{Mg Naing Lin,Candycrushsodasaga}} = \frac{\sum_{N \in \text{similarTo i(Candaycrushsodasaga)}} (S_{\text{Candyctushsodasaga,N}} * 0)}{\sum_{N \in \text{similarTo i(Candaycrushsodasaga)}} (|S_{\text{Candycrushsodasaga,N}}|)}$$

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

$$= -0.58$$

3.2.5 De-normalization

Finally the system predicts how will the current user rate application2 and application7. The results are calculated by using equation (3.5) is de-normalization into rating format in the range (1 to 5)

$$R_{u,N} = \frac{1}{2}((NR_{u,N} + 1) * (Max_R - Min_R)) + Min_R \qquad (3.5)$$

Where,

$R_{u,N}$ 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 3.4 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	Mg Aung Myat	5	4	3	2	1	5	3	-	-	-	1	2.24
2	Mg Naing Lin	3	2	2	1	2	3	1	-	-	-	2	1.74
3	Ma Phoo Pwint Thazin	5	4	3	2	1	2	2	-	-	-	3	2.52
4	Ma Yoon Nadi Zaw	2	1	2	3	3	2	1	-	-	-	2	2..02
5	Mg Myo lin	5	2	1	3	2	1	2	-	-	-	1	1.90
6	Mg Aung Bo Bo Zaw	4	3	2	1	2	3	2	-	-	-	3	2.45
7	Ma Khin Htet Htet Kyaw	5	4	3	2	1	2	3	-	-	-	3	2.07
8	Ma Ei Thazin Phyo	5	2	1	3	2	1	3	-	-	-	1	1.90
9	Ma Khin Myat Thu	5	2	1	1	2	1	2	-	-	-	5	2.00
10	Ma Su Myat Wai	5	4	3	2	1	5	3	-	-	-	1	2.24
11	Ma Hla Hla Aye	3	2	2	1	2	3	1	-	-	-	2	1.79
12	Ma Tin Zar Wine	5	4	3	2	1	2	2	-	-	-	3	2.52
13	Ma Aye Myat Thu	2	1	2	3	3	2	1	-	-	-	2	2.02
14	Mg Myo Thant Zaw	5	2	1	3	1	3	2	-	-	-	1	1.90
15	Ma Khin Myo Thant	4	3	2	1	2	3	2	-	-	-	3	2.45
16	Mg Kyaw Myint	5	4	3	2	1	2	3	-	-	-	3	2.07
17	Ma Thuzar Wai	5	4	3	2	1	5	3	-	-	-	1	2.24
18	Mg Zaw Mn Naing	3	2	2	1	2	3	1	-	-	-	2	1.79
19	Mg Win Mg Oo	5	4	3	2	2	2	2	-	-	-	3	2.52
20	Ma Khin Thu zar	2	1	2	3	3	2	2	-	-	-	2	2.00

CHAPTER 4

SYSTEM DESIGN AND IMPLEMENTATION

4.1 Components of the System

The recommender system of the mobile phone applications needs to collect data. These applications are downloaded from apkpure.com, Google play store and Microsoft Office.com. This application is collected by name, version, size, and source. The collected data can see as the following list. The collected data and rating values are obtained by surveying. There are two main components of this recommender system: admin and user. The functions of these components are detailed presented as follows.

4.1.1 Functions of the Admin

When the admin enters into the system, he/she must login using his/her username and password into this system. If these username and password are valid, he/she will be permitted to login into this system.

At first step, the admin can add new application and it consists of information: version, name, source, and size. After that, the admin can view the applications list that has been added before. When the added applications' information is incorrect, they can be deleted and edited.

Moreover, the admin can look at in rating list for the items that has been rated by the users. The system calculates that the average based on the rating values that are given by the users.

As the second step, this system groups the users into clusters according to their age and average rating values and calculates similarity values based on average rating values that have been calculated in previous step in specific clusters. This system uses the adjusted cosine similarity to find the similar items.

As the third step, this system calculates normalization, prediction, de-normalization based on similarity values. Finally, the system gives the predicted rating value and recompute average rating and similarity values. Then, the system can recommend the similar applications with the user' wants.

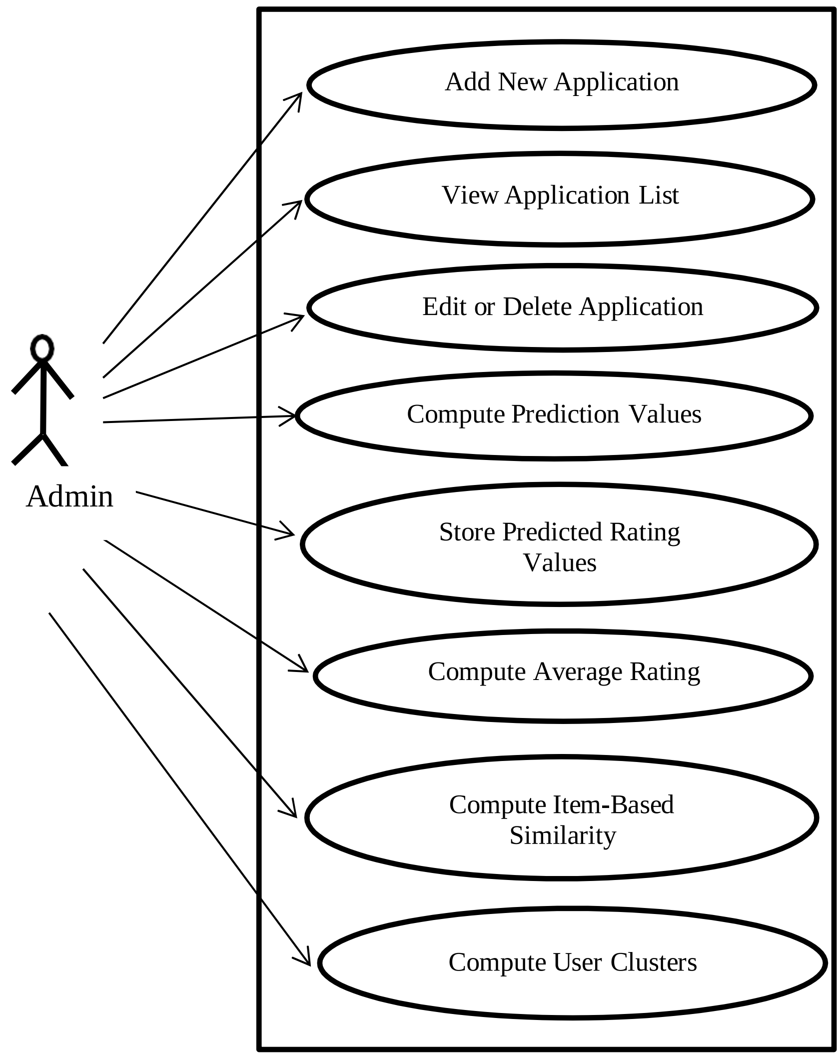

Figure 4.1 Use Case Diagram for Admin

4.1.2 Functions of the User

If the user wants to enter into the system, he/she must register as a member. If the user is a member, he/she can login by using username and password. If the username and password are correct, the user can view applications list and search the interested applications. There are three types of searching methods: all applications, keyword, application name and source. Then, the user can see detail information of the application that system recommends. The user can also download the application if he wants to it. If he/she wants to give rating value for this application, he/she can give rating scale between 1 and 5.

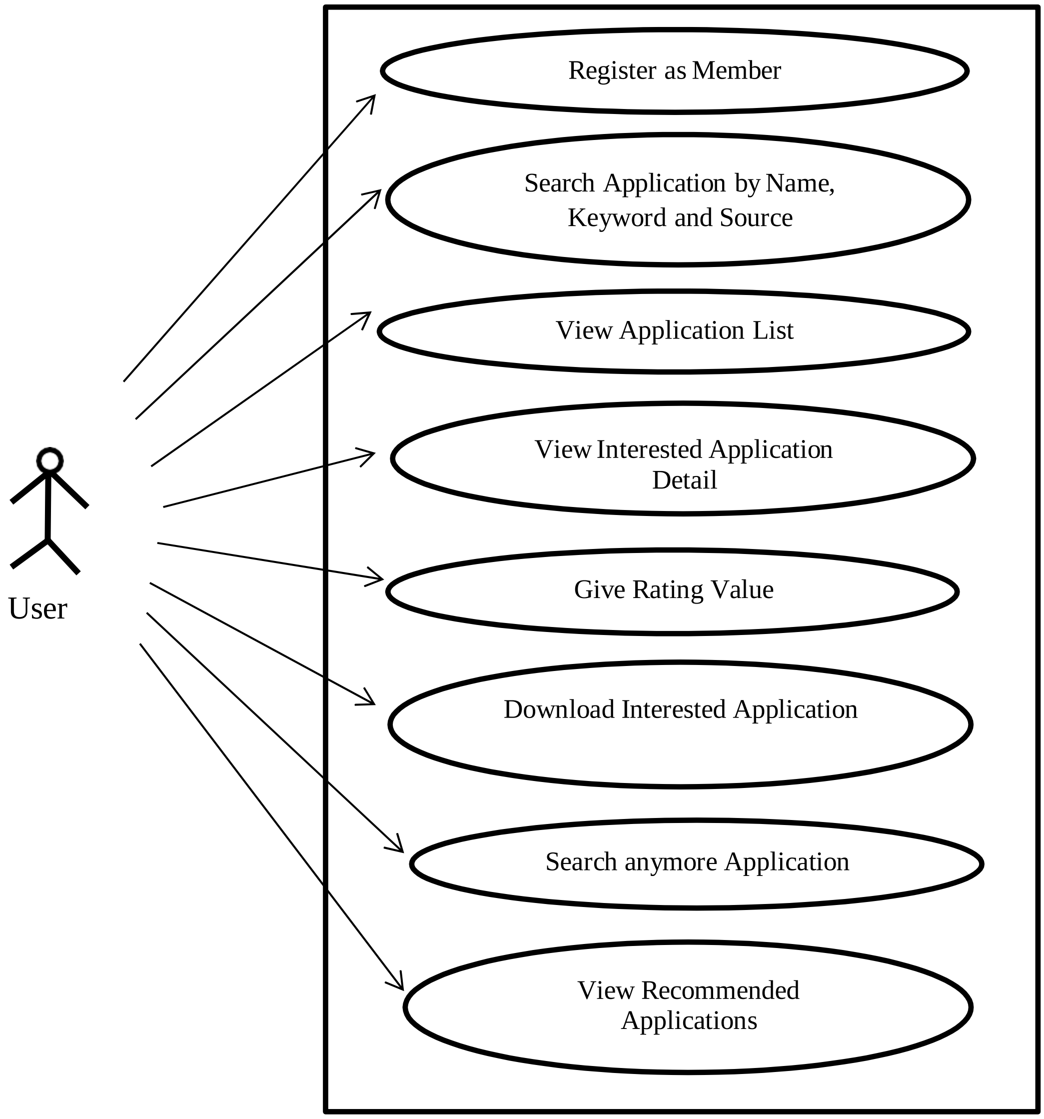

Figure 4.2 Use Case Diagram of Users

4.2 System Flow

There are different two types of system flow in this system. They are system flow for admin and system flow for user.

4.2.1 System Flow for Admin

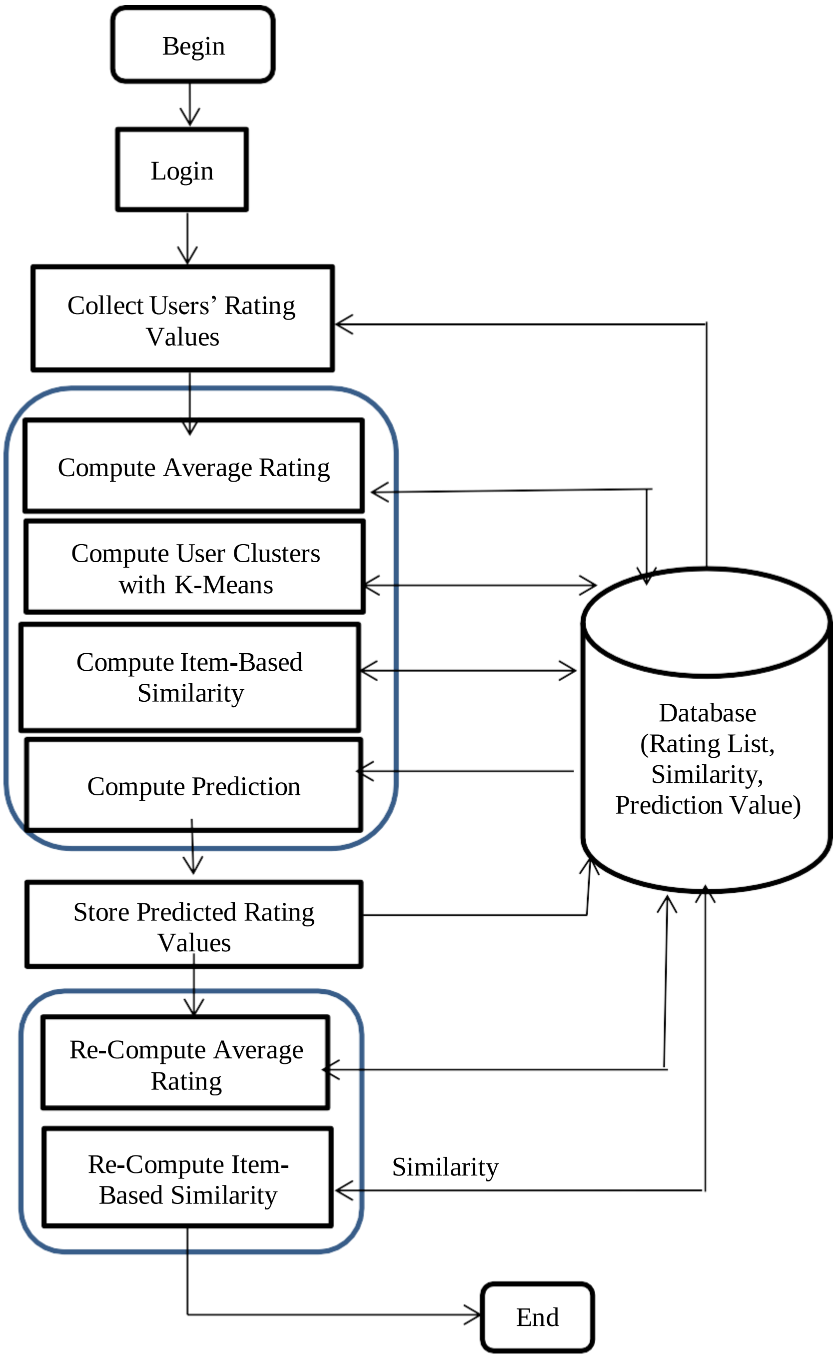

Figure 4.3 System Flow Diagram for Admin

After the admin have done login, he/she collects users' rating value from the database and average rating is calculated based on rating values. Then, the system groups the users clusters with K-means clustering. The system will generate similarity matrix based on item-based collaborative filtering technique by using adjusted-cosine similarity. Normalized rating and prediction using weighted sum will be calculated. And then de-normalization step is used to predict the rating values which was not given rating by the user. The predicted rating values are stored into database.

Moreover, the admin recomputes average rating and simalarity with predicted rating values. Finally, the system gives the similar items to the user as recommended applications.

4.2.2 System Flow for User

The user needs to login into this system. If he/ she isn't already member, he/she needs to make a registration. If he/ she is already member, he/she can search the interested applications using with criteria: name, keyword, and source. Moreover, they can view the application list and detail respectively. The user can give or not the rating value between (1 to 5). After the user has given the rating value, he/she can see the recommended applications. The users can make searching anymore by their interested application if they want.

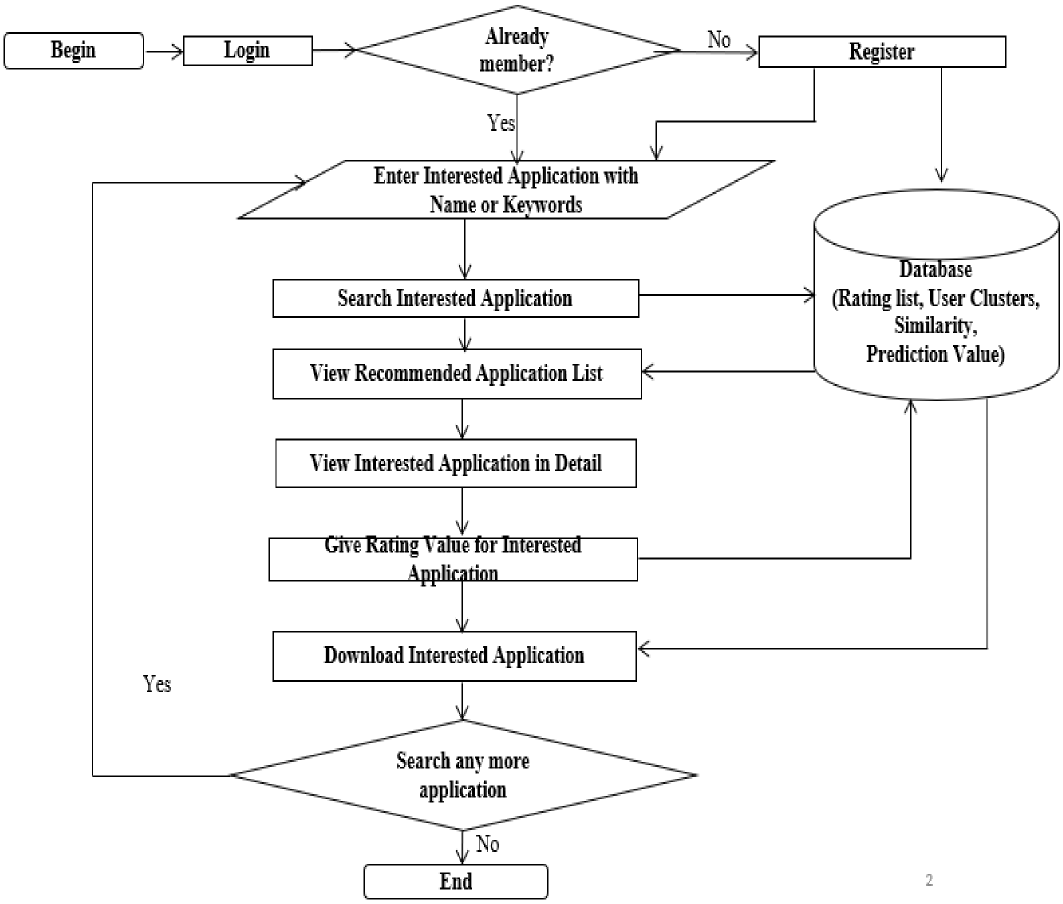

Figure 4.4 System Flow Diagram for User

4.3 Evaluation Metrics

The section describes the evaluation metrics that are widely used in recommender systems. The performance of the proposed system will be shown in terms of precision and recall. Precision and recall that are the most popular metrics used for evaluating information retrieval systems.

4.2 Precision of the System

In this system, the ratio of the retrieved applications that are relevant and the number of all retrieved applications can be defined as precision.

e.g, the proportion of recommended applications that are actually relevant

$$Precision = \frac{Recommended\ relevant\ applications}{All\ Recommended\ Applications} \tag{4.1}$$

Precision = 17 / 21 = 0.8095 * 100 = 80.95 %

4.2.3 Recall of the System

In this system, the ratio of the relevant recommended and all relevant applications can be defined as recall. 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

$$Recall = \frac{Recommended\ Relevant\ Applications}{All\ Relevant\ Applications} \tag{4.2}$$

Recall = 17 / 17 = 1 * 100 = 100 %

4.3 Experimental Results

The section presents the experimental results of the proposed system. In these results, the precision and recall values are computed depending on 64 applications and 26 users. Table 4.1 shows the results in detail.

Table 4.1 Experimental Results

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
6	Block Puzzle Conquer	17	23	17	73.91	100
7	Blossom Blast Saga	21	24	21	87.50	100
8	Bomb Squad	15	21	15	71.43	100
9	Mini World Block Art	16	18	16	88.89	100
10	Fruits Legend	21	22	21	95.45	100

11	Bubble Shooter Genies	14	17	14	82.35	100
12	Bubble Shooter	14	18	14	77.78	100
13	Bubble Witch 2 Saga	16	17	16	94.12	100
14	Bunny Blast	18	25	18	72.00	100
15	Candy Crush Friends	20	25	20	80.00	100
16	Candy Crush Saga	15	23	15	65.22	100
17	Candy Fever	19	22	19	86.36	100
18	Car Merger	15	19	15	78.95	100
19	Clash of Clans	21	25	21	84.00	100
20	Dragon Merger	18	20	18	90.00	100
21	Hay Day	16	21	16	76.19	100
22	Farm Heroes Saga	12	18	12	66.67	100
23	Fruits Bomb	16	20	16	80.00	100
24	Genies Gems	20	24	20	83.33	100
25	Gems Jewel Crush Match 3	20	24	20	83.33	100
26	Snoopy Pop	13	15	13	86.67	100
27	Logic Master 1 Mind Twist	14	15	14	93.33	100
29	Terrarium Garden Idle	19	22	19	86.36	100
30	SpeedBall	20	22	20	90.91	100
31	Subway Surfers	20	22	20	90.91	100
32	Power Painter	17	19	17	89.47	100
33	Panda Pop Bubble Shooter Game Blast Shoot Free	20	26	20	76.92	100
34	My Heroes Dungeon Adventure	22	25	22	88.00	100
35	slither io	23	25	23	92.00	100
36	Sweet Fruit Candy	21	25	21	84.00	100
37	Word Search Games in English	14	18	14	77.78	100
38	Ludo Master New Ludo Game 2018 For Free	13	15	13	86.67	100

39	Matches Puzzle Game	13	18	13	72.22	100
40	Merge All	13	20	13	65.00	100
41	Jewels classic Prince	11	16	11	68.75	100
42	Jewels Jungle Match 3 Puzzle	16	20	16	80.00	100
	Average				**81.31**	**100**

CHAPTER 5
CONCLUSION, LIMITATIONS AND FURTHER EXTENSIONS

In this chapter, the system is concluded, the limitation and the further extensions of this system are also presented. This recommendation system is to use K-Means Clustering and Item-based Collaborative Filtering.

5.1 Conclusion of the System

The proposed system helps the users to search their interested mobile applications that they want to download or would like to view. This system applies the K-Means clustering to cluster the users based on their age and rating styles and item-based collaborative filtering to find similar applications. By using the clustering step, this system will provide more useful and effective recommendation results about mobile applications for mobile phone users than the traditional item-based collaborative filtering. As future work, there is necessary to develop a novel recommendation system in large dataset with more applications and users rating values. The proposed system will also apply in other applications such as e-commerce sites, etc.

5.2 Limitations

In this system, there are some limitations. This system cannot perform the similarity value if user has not rate items because this system is based on item-based rating methods. Moreover, this system cannot solve the wrong spelling problem in searching applications.

5.3 Further Extensions

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. This system can be compared with user based collaborative filtering method.

REFERENCES

[1] Arissa Aprillia Nurcahyani " Recommendation Systems for Mobile Phone Device Application Downloading Using Item Collaborative Filtering Method Implementation", 2009.

[2] Aye Aye Lwin "Comparing Item-Based to User-Based Approach of Collaborative Filtering Methods", March, 2017.

[3] Hanafi, Nanna Suryana, Abdul Samad Bin Hasan Bashari "Collaborative Filtering Implementation In Recommender System" Vol.95, No.16, 31 Aug 2017, Malaysia.

[4] Htin Lin Aung "Personalized Book Recommendation System By Usig Neighbourhood-Based Method", 2012.

[5] "Implicit ratings and item based filtering" ,China.

[6] Jennifer Pahlka "Neighborhood- Based Collaborative Filtering" 2016, Switzerland.

[7] Kothuri Sravya Sri Poornima, N.Vijaya Kumar, "Item Based Collaborative Filtering Recommendation Algorithm", Vol. 06, Issue.01, June 2018, India.

[8] Mya Thu Zar "Book Recommender System Using Collaborative Filltering and Content-Based Techniques", Oct, 2015.

[9] 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

[10] Taner Arsan, Efecan Koksal, Zeki Bozkus "Comparison of Collaborative Filtering Algorithms With Various Similarity Measures For Movie Recommendation" Vol.6, No.3, June 2016, Turkey.

[11] Zaw Lin Htay "Online Car Recommendation Using Collaborative Filtering-Based Agoregated Query (CFAg Query), May,2016.

[12] Khin Mar Cho, Mya Sandar Kyin, “Mobile Game Applications Recommendation System with Item-Based Collaborative Filtering”, IJARND, 2020

[13] Anand Shanker Tewari, “Generating Items Recommendations by Fusing Content and User-Item based Collaborative Filtering”, ICCIDS 2019

[14] Chigozirim Ajaegbu, “An optimized item-based collaborative filtering algorithm”, JAIHC 2020
[15] Galyna Chornous, “A hybrid user-item-based collaborative filtering model for e-commerce recommendations”, JIS 2021
[16] Nadia Fadhil AL-Barkri, Soukaena Hassan Hash Hashim, “Collaborative Filtering Recommendation Model Based on K-Means Clustering”, A1-Naharain Journal of Science, 2019
[17] Yuhangzi Fu and Tian Wang, “Item-Based Collaborative Filtering with BERT”, Proceedings of the 3rd Workshop on e-Commerce and NLB (ECNLP 3), 2020
[18] mohd Abdul Hameed, Omar A1 Jadaan and S. Ramachandram, “Collaborative Filtering Based Recommendation Syste: A Survey”, IJCSE 2012
[19] Robin Burke, Alexandar Felfernig, “Recommender Systems: An Overview”, Advancement of Artificial Intelligence, 2011
[20] Mohamad Ivan Fanany, “Introduction to Recommender System”, 2015
[21] Liaoliang Jiang, Yuting Cheng, “A trust-based collaborative filtering algorithm for E-commerce recommendation system”, Springer, 2018
[22] Leandro Krug Wives, Valter Roesler, “Enhancing the accuracy of ratings predictions of video recommender system by segments of interest”, 2013
[23] Bo Hung, Mengchen Yu, “A collaborative filtering algorithm based on correlation coefficient”, Springer 2019
[24] Jing Y, Zhang L, Phelan C, “A novel recommendation strategy for user-based collaborative filtering in intelligent marketing”, 2016
[25] https://www.amazon.com
[26] https://www.netflix.com
[27] https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol
[28] https://www.hostinger.com/tutorials/what-is-html
[29] https://towardsdatascience.com/various-implementations-of-collaborative-filtering-100385c6dfe0
[30] https://medium.com/@atharvanchalke/content-based-filtering-and-collaborative-filtering-9bf006810618
[31] https://en.wikipedia.org/wiki/Internet
[32] https://en.wikipedia.org/wiki/World_Wide_Web

APPENDIX

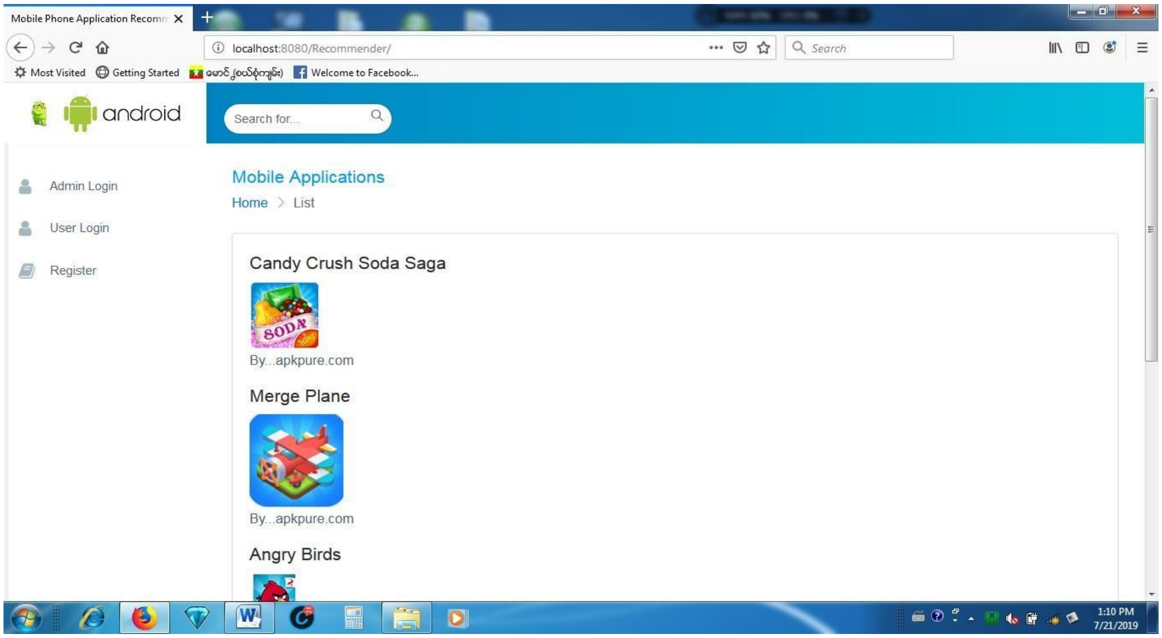

Figure A1. Recommender Web Page

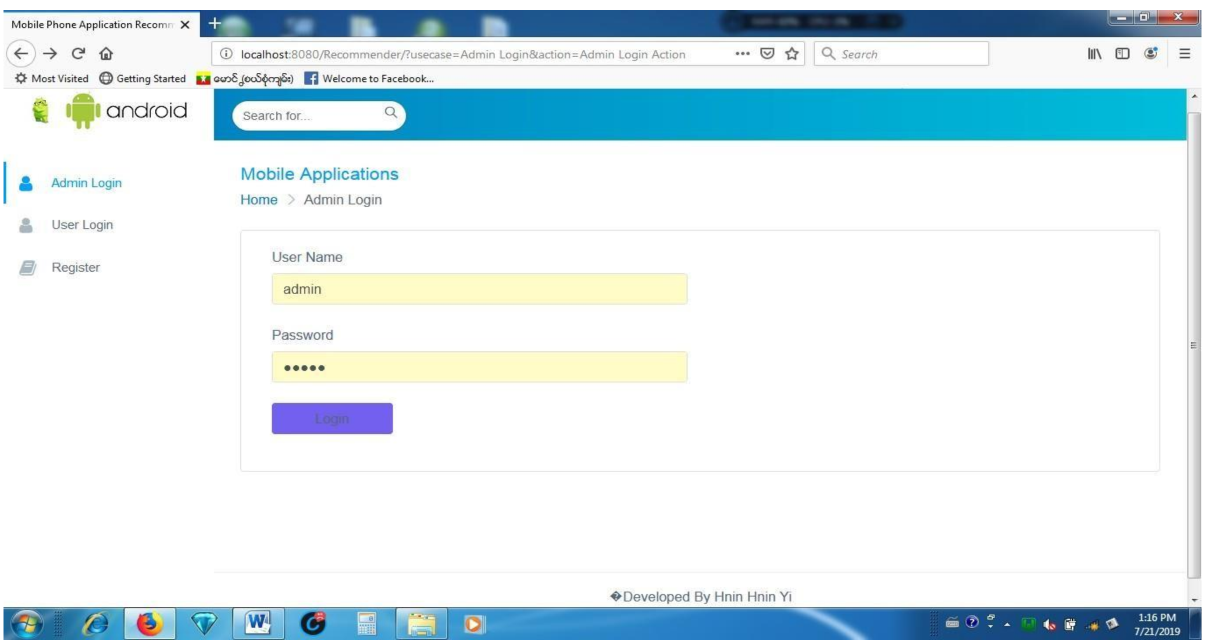

Figure A2. Admin Login Page

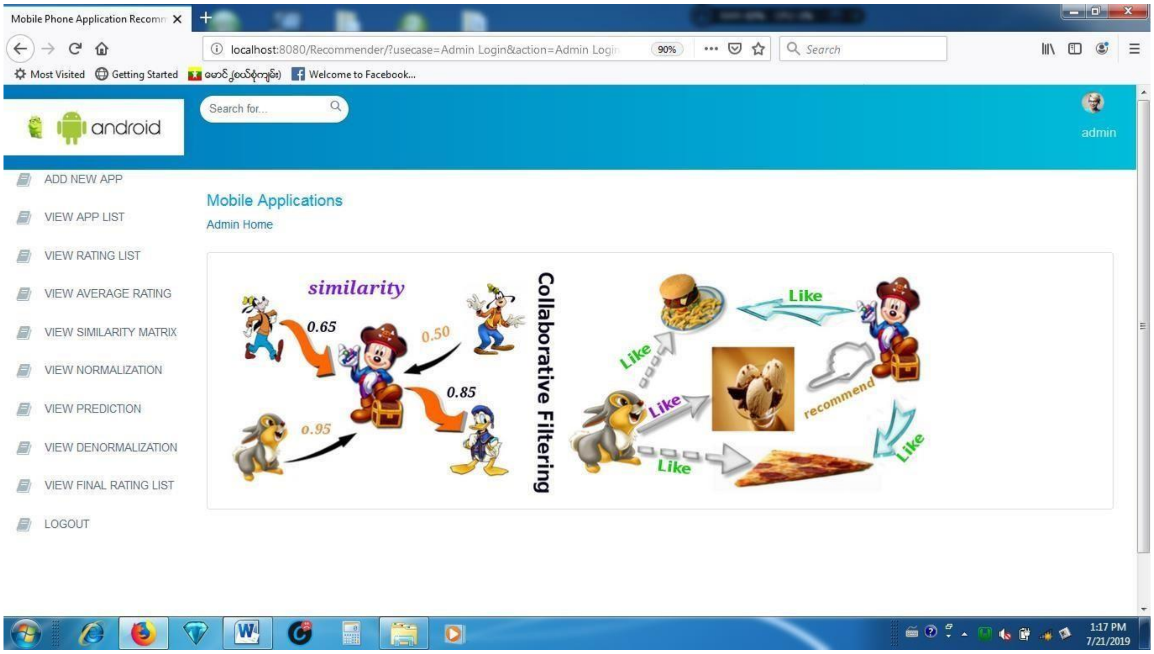

Figure A3. Admin Home Pages

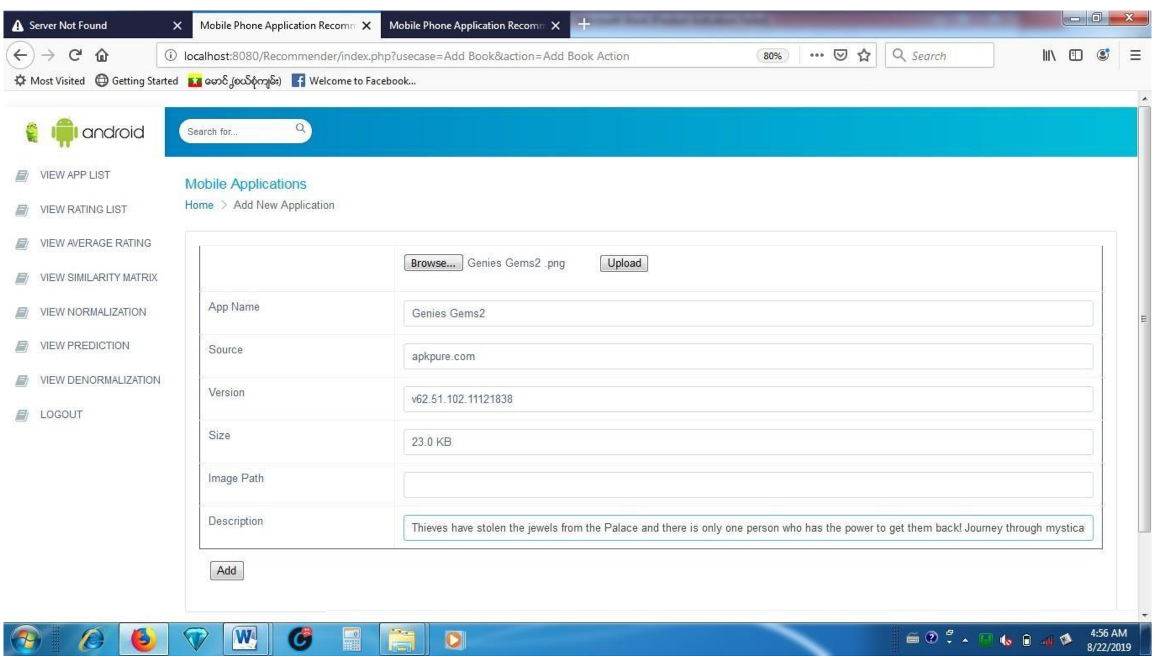

Figure A4. Add new application page

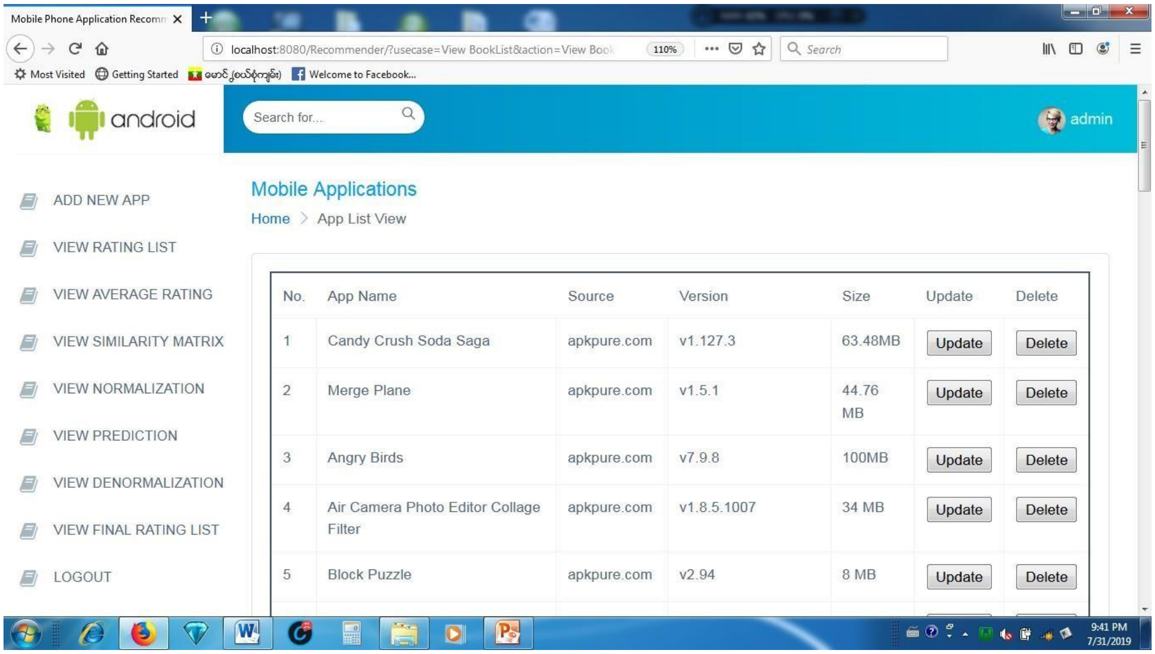

Figure A5.View Application List Page

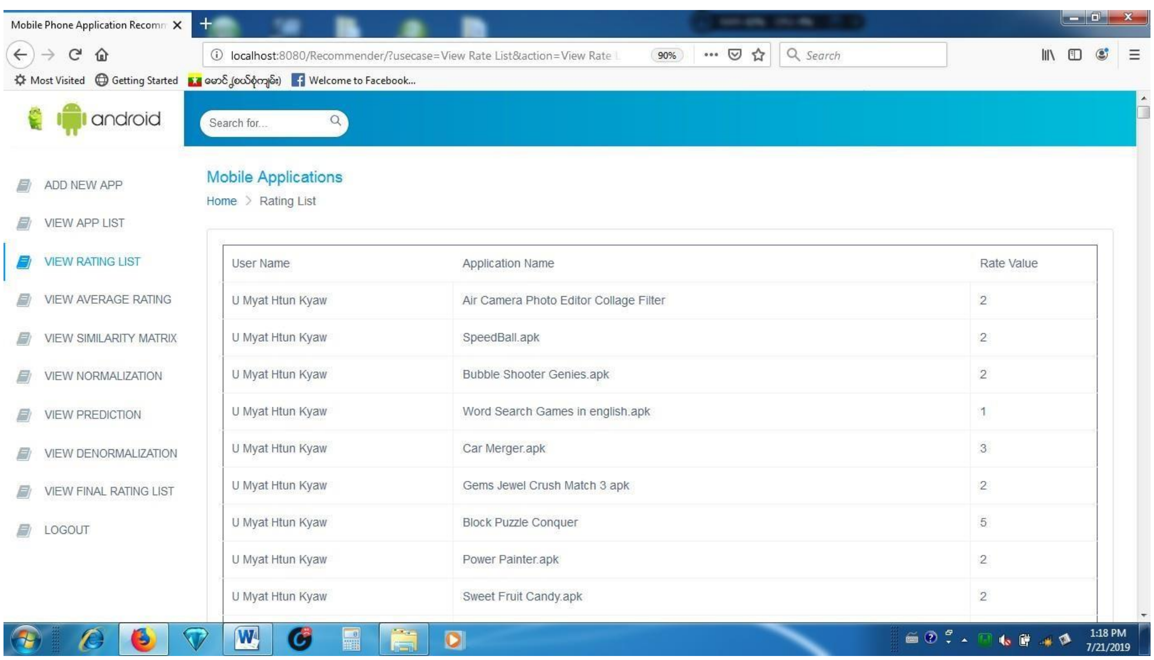

Figure A6. View Rating List

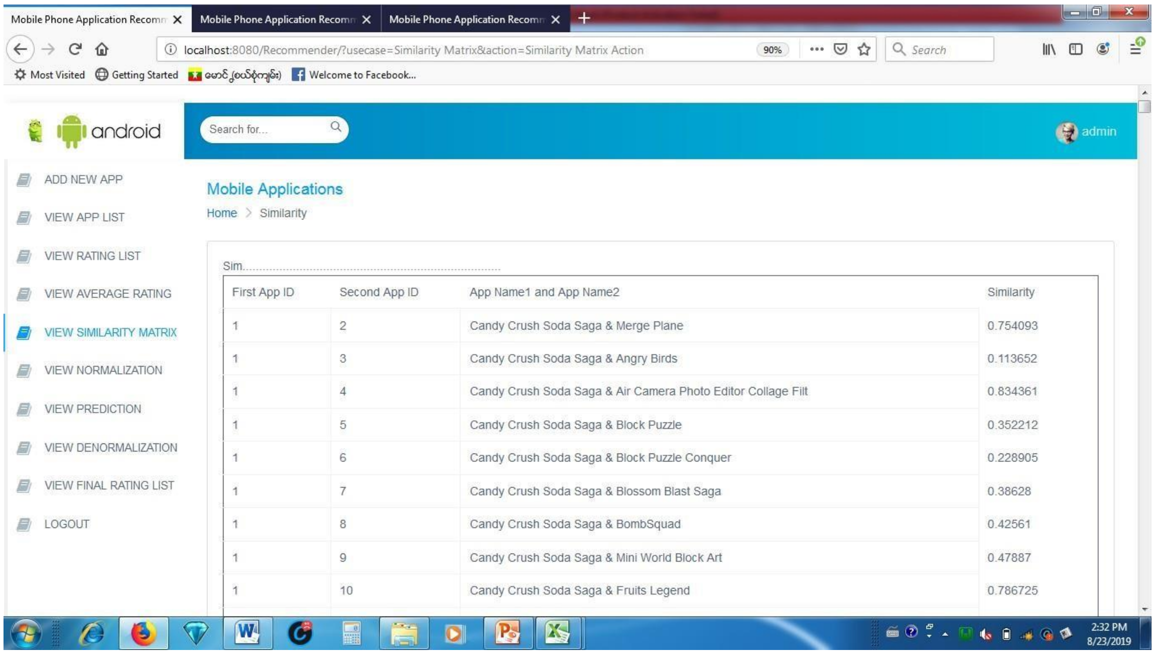

Figure A7. View Similarity Values

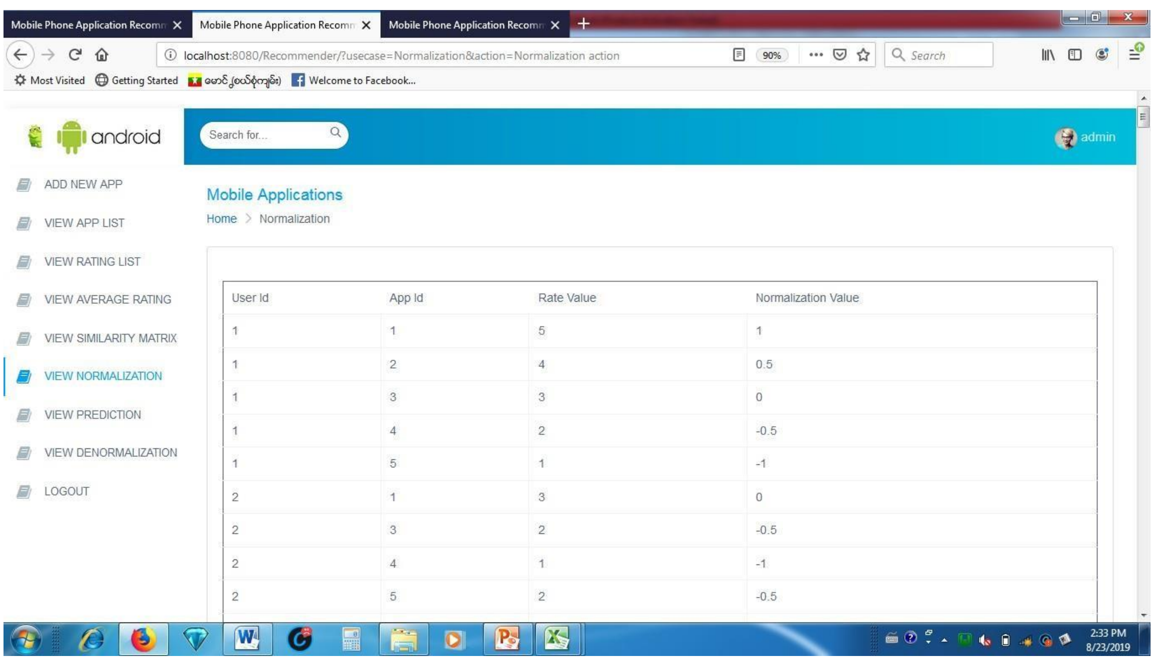

Figure A8. View Normalization Values

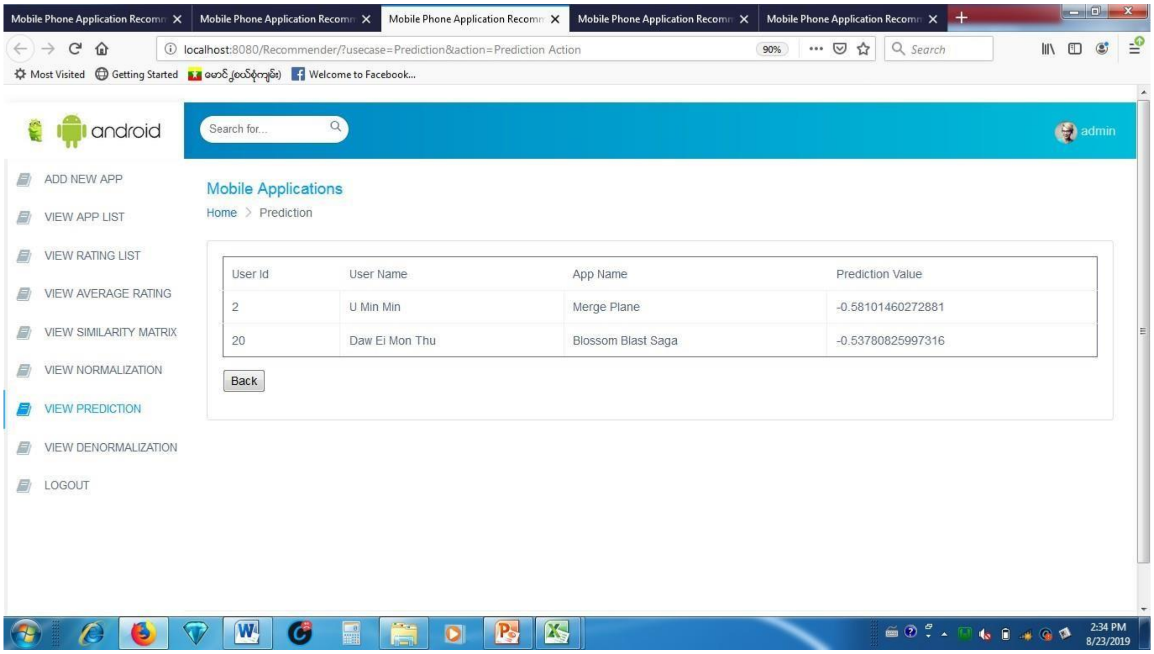

Figure A9. View Prediction Values

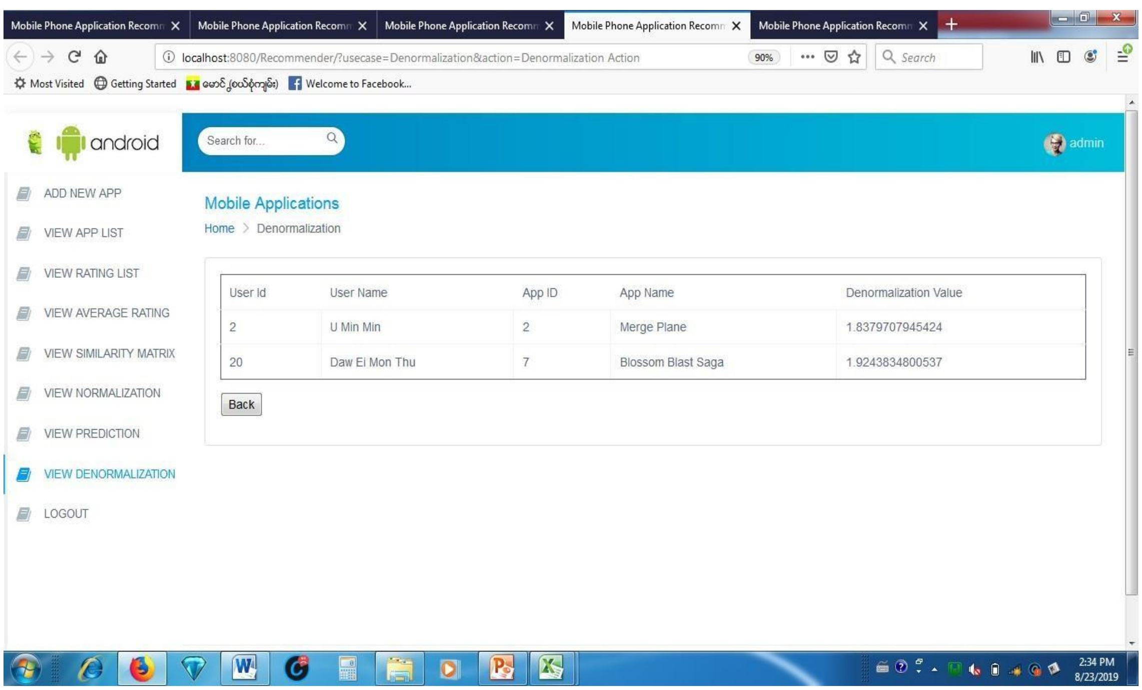

Figure A10. View De-normalization Values

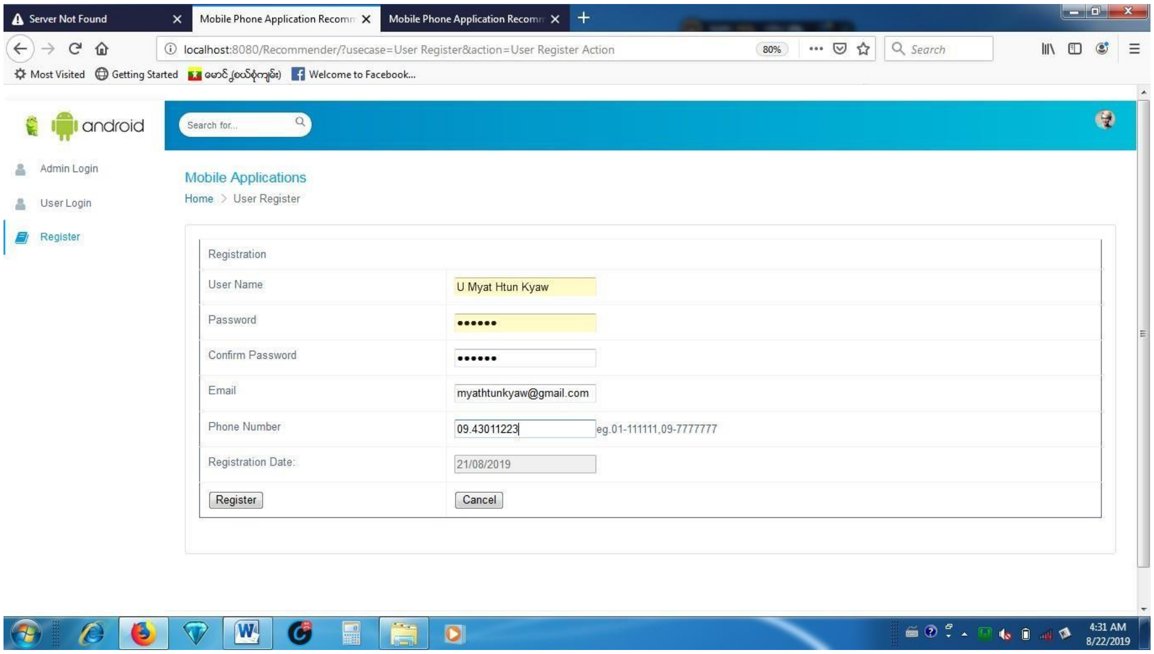

Figure A11. User Register Page

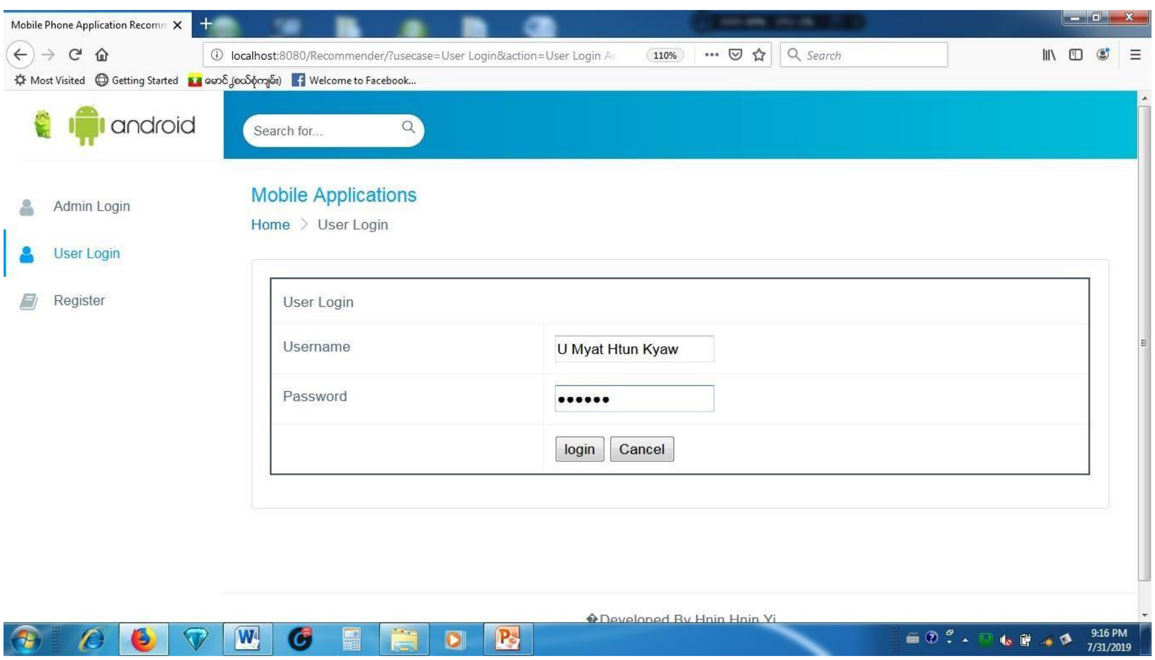

Figure A12. User Login Page

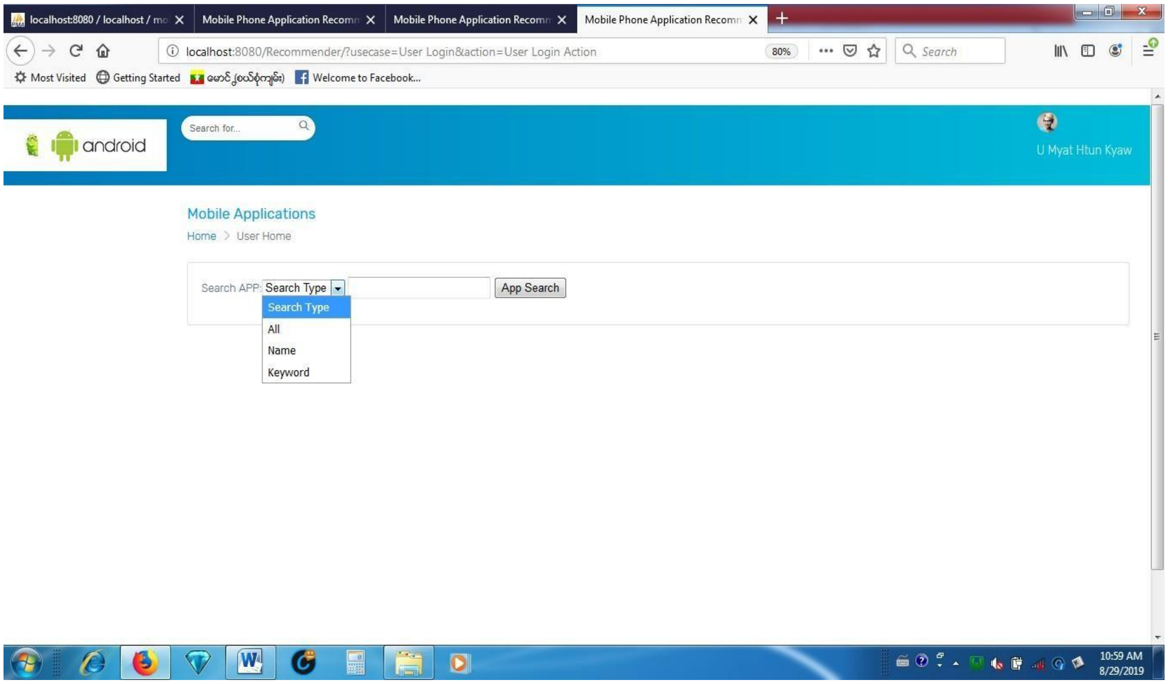

Figure A13. User Home Page

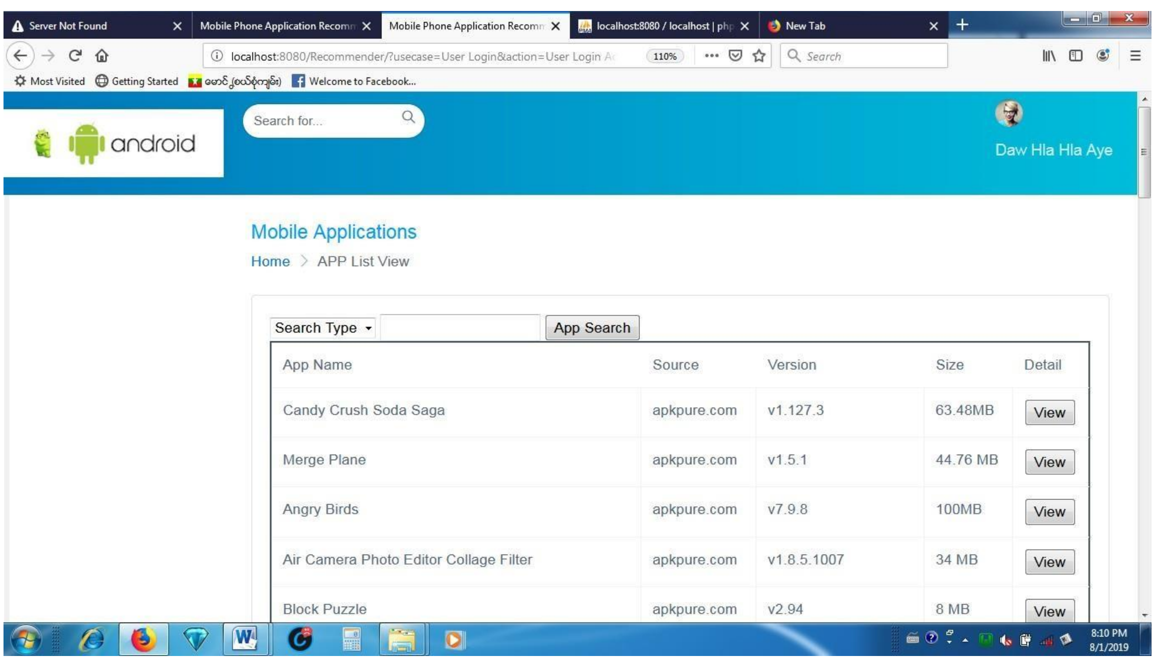

Figure A14. View Application List

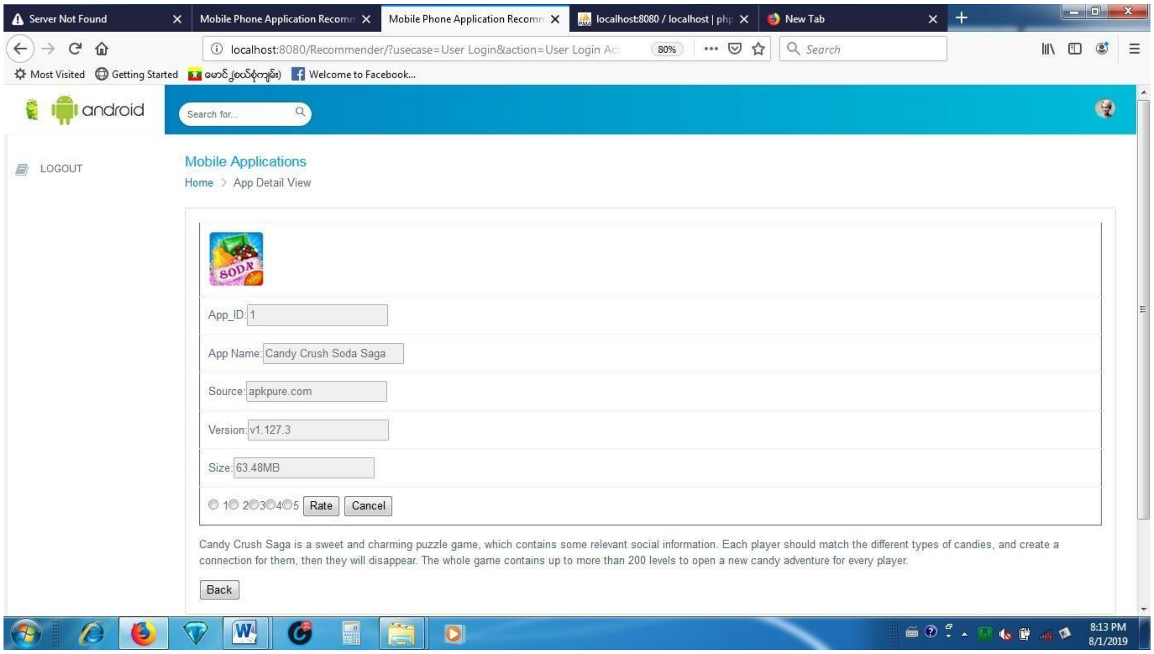

Figure A15. View Application Detail

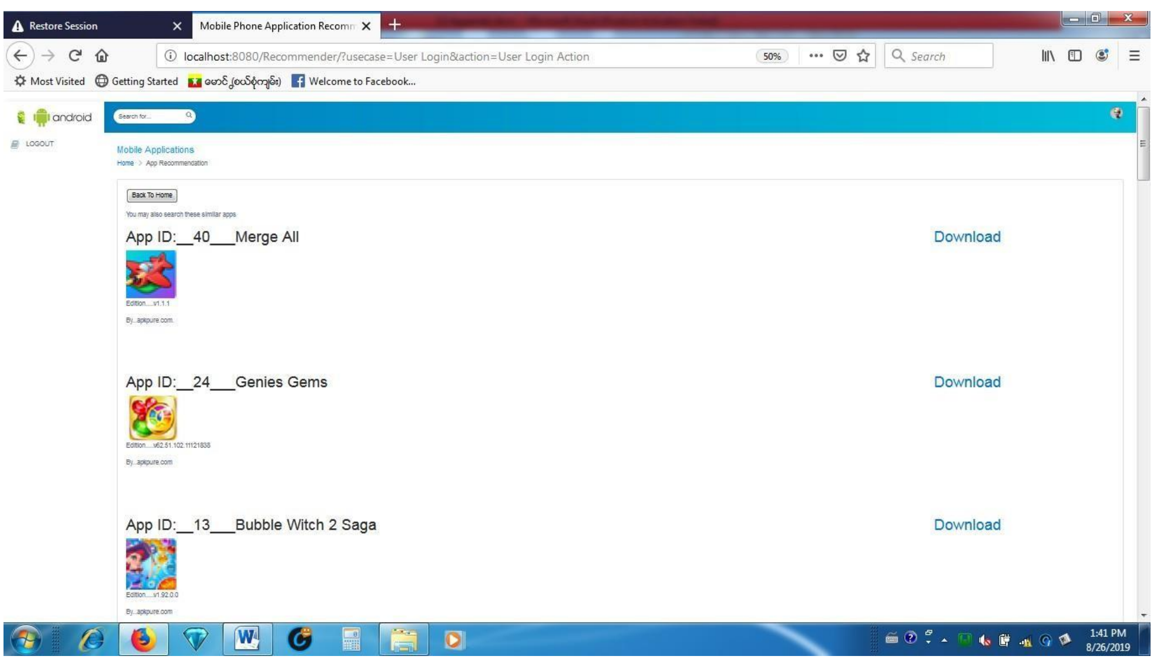

Figure A16. View Recommended Application