Top-k Recommended Items: Applying Clustering Technique for Recommendation

Kittisak Onuean[1], **Sunantha Sodsee**[2], and **Phayung Meesad**[3]

ABSTRACT

This research proposes the Top-k Items Recommendation System which uses clustering techniques based on memory-based collaborative filtering technique. Currently, data sparsity and quantity of system are problems in memory-based collaborative filtering technique. We offer recommend or show some items set for user's preference. In this research, we propose methods for recommended items set to user preference on data sparsity, movie lens datasets (1M) consisting of 671 users and 163,949 product items were used by determining the preference level between 1 and 5 and user satisfaction levels of all 98,903 items being build and test the models. Methods was divided into three parts included 1) Simple Agent Module 2) Neighbor Filtering and 3) Prediction and Recommend. Simple clustering was used to create a system to provide suggestions for sparsity data. Datasets obtained from clustering represented the sample agent of dataset to being create the recommendation system. Datasets were divided into two categories, 1) Traditional Data (TD) and 2) Statistic Data (SD), and each dataset clustered by k-means clustering. The experimental results demonstrated that the number of item types in the system were recommended in the TD and Euclidean (DIS). DIS was used to find the nearest value in TD for the item list recommendation to active users in the system with the a lot of number choice of recommendation system.

Keywords: Recommender System, Clustering Technique, Data Sparsity

1. INTRODUCTION

Recommendation systems have been developed in the past by using clustering techniques [1]. Presently, recommendation systems are popularly used to generate lists of products or services for system users. The main processes to create a successful recommendation system consists of finding similarity, screening population, predicting and providing an accurate recommendation. Different algorithms are used to calculate similarities. A calculation is based on the recommendation system developed by rating a preference level for a similar product. To develop the system, the main procedures employ finding similarities between users, this will be used to find a user group who needs data for the prediction (Top-k). In determining the similarity, the usage of high-dimensional data affects the efficiency of calculating the similarity and the recommendation system. Presently, several researchers have developed and improved the data dimension issue to support high-dimensional data to calculate the similarity of data in order to gain a greater efficient recommendation system.

Thus, this research proposes the Top-k Items Recommendation System which uses clustering techniques for recommendations. A major procedure is k-means clustering, which is used to cluster a user group, the dataset will be used to create the recommendation system. The main procedures are finding the similarity, prediction, providing recommendation, and testing the efficiency of the recommendation system.

This research applies a technique to the Top-k Recommended Items System which supports sparse data [2,3] by using K-means clustering to improve the recommendation system. The objectives are 1) to propose a selection technique for the Top-k Items Recommendation System for recommendations, and 2) to evaluate the efficiency of the Top-k Items Recommendation System. The aim of this research was to propose methods for recommended items set for the active user having data sparsity problem. We aimed our developed method could recommend items based on simple approach and supported to general systems.

2. BACKGROUND AND LITERATURE REVIEW

2.1 Recommendation System Overview

A collaborative filtering technique (CF) [1, 4] is commonly applied to create a recommendation system. By collecting the data of users' interest in products, a recommendation system is created to introduce interesting products for active users. The processes are as follows: 1) Specify the number of users similar to present users. 2) Apply a rating scale method to similar products.3) Continue from step 2

Manuscript received on June 25, 2018 ; revised on December 1, 2018.

Final manuscript received on January 30, 2019.

[1] The author is PHD student with Faculty of Information Technology, King Mongkuts University of Technology North Bangkok,10800, Thailand., E-mail: on.kittisak@gmail.com

[2,3] The authors are with Faculty of Information Technology, King Mongkut's University of Technology North Bangkok,10800, Thailand., E-mail: sunanthas@kmutnb.ac.th and phayung.m@it.kmutnb.ac.th

and select the products for recommendation with the following technique.

The memory-based Collaborative Filtering Technique is a technique to apply data of users' histories related to their products of interest. It also contributes to a illustrating dimension table with ranks of products of users who feel interested. Then, the similarity between new and former users within the system or between new and existing products in the system will be calculated. After that, the users or products which expose the high degree of similarity are predicted to obtain N sets of data and to give advices. The limitation of this method is data sparsity.

Model-based Collaborative Filtering Technique is a technique that applies machine learning algorithms to create a recommendation system. It is used when a new user enters the system when none have been before. The most common algorithms applied to create recommendation systems are: Bayesian classifiers, neural networks, fuzzy systems, genetic algorithms, Latent Features, and matrix factorization, etc.

The Hybrid Collaborative Filtering Technique is the combination of collaborative filtering and graphic filtering or the combination of collaborative filtering and content-based filtering to improve the effectiveness of recommendation systems.

2.2 Similarity Computation

The similarity value [1,4,5] of creating a recommendation system by applying the collaborative filtering technique on the basis of need is calculated by using users' histories of products of interest to calculate the similarity value of the active users who recently enter the system. Researchers have created recommendation systems by applying the algorithms: Pearson's Correlation (COR), Cosine (COS) and Adjusted Cosine (ACOS) for similarity between, etc. Researchers often apply each type of algorithm to find the similarity between former and active users in the system. Then, they normally try to find neighborhoods to use as ranking data to predict a level of active users' interest towards the products. In this research, the methods applied to find the similarity are as follows:

2.2.1 Pearson's Correlation (COR)

The determination of the similarity between u_a and u_b users,$r_{u,i}$ is specified as the preference of products i of user u ,and $\bar{r}$ indicates a mean score of the overall user u's preference that is ranked the same as users (co-rate). Moreover, n refers to the number of products ranked at the same level by users (co-rate).

$$sim(u_a,u_b)=\frac{\sum_{h=1}^{n}(r_{u_a,i_h}-\bar{r}_{u_a})(r_{u_b,i_h}-\bar{r}_{u_b})}{\sqrt{\sum_{h=1}^{n}(r_{u_a,i_h}-\bar{r}_{u_a})^2}\sqrt{\sum_{h=1}^{n}(r_{u_b,i_h}-\bar{r}_{u_b})^2}} \tag{1}$$

2.2.2 Cosine (COS)

The cosine is evaluated by similarity dimension value that uses $r_{u,i}$ as the level of the product i preference. User u and n indicates the number of products that is co-rated by the users.

$$sim(u_a,u_b)=\frac{\sum_{h=1}^{n}(r_{u_a,i_h})(r_{u_b,i_h})}{\sqrt{\sum_{h=1}^{n}r_{u_a,i_h}^2}\sqrt{\sum_{h=1}^{n}r_{u_b,i_h}^2}} \tag{2}$$

2.2.3 Euclidean [6,7]

The Euclidean is a technique to find the similarity between user u_a and u_b to rate the score of preference for product i. Less value of the difference implies a high level of the similarity

$$Distanct(u_a,u_b)=\sqrt{\sum_{i=1}^{n}(r_{u_a,i}-r_{u_b,i})^2} \tag{3}$$

As a result, the new equation to find similarity value is as follows [8]

$$sim(u_a,u_b)=\frac{1}{1+Distanct(u_a,u_b)} \tag{4}$$

2.2.4 Jaccard (JACC)

The Jaccard is a method to find the similarity. I_{u_a} is used as the data set of products that are rated in accordance with the importance by user u_a.

$$S(u_a,u_b)=\frac{Iu_a \cap Iu_b}{Iu_a \cup Iu_b} \tag{5}$$

Apart from the equations mentioned above, a few researchers have applied and developed equations to find the similarity value in different forms, shown in the following section.

2.3 Neighbors Selection

The neighbors selection algorithm can be applied to make predictions. A few researchers have divided the data selection methods into 2 types to predict active users [9], namely 1) Nearest Neighbors algorithms: These methods are employed to find the similarity of new and current users by collecting the existing data to find the similarity of the active users because the active users might have the same product preference with existing users based on the data history. These algorithms can be divided into 2 categories in terms of users' perspectives and in terms of the products and active users in the system. 2) Top-k recommendation: this method applies the data to make predictions by configuring k as the number of users' data sets or products with a high level of the similarity analyzed from a dimension table of users-products by finding the correlation between users or products and products. Then, the data will be calculated to find a level of the similarity of the active users by selecting the users derived from the nearest neighbor before making a prediction and a recommendation computation.

Prediction and Recommendation Computation are regarded as a crucial part to introduce and rank based on a level of interest of the active users who newly enter into the system. There are 2 well-known methods in these algorithms. Firstly, Weighted Sum of Others': this method is a prediction of a rank of interest by using the weighted sum of rank of interest. Another method is Simple Weighted Average, which is a prediction of rank of interest rated by applying a mean score.

2.3.1 Weighted Sum of Others' Ratings (WS)

$$Pu_a, i_a = \bar{r}_{u_a} + \frac{\sum_{h=1}^{n} (r_{u_h,i_a} - \bar{r}_{u_h}) \cdot S(u_a, u_h)}{\sum_{h=1}^{n} |S(u_a, u_h)|} \quad (6)$$

From equation (6) above, $\bar{r}_{u_a}$ is used as the average score of user a's preference for all products n is applied as the number of users who rate the score for product i_a, and $S(u_a, u_h)$ value is the similarity between user a and user h. The calculation is made to all members who rate the score for product i_a in order to make a prediction on the basis of the nearby neighbors of new incoming users.

2.3.2 Simple Weighted Average (SWA)

$$Pu_a, i_a = \frac{\sum_{h=1}^{n} r_{u_a,i_h} \cdot S(i_a, i_h)}{\sum_{h=1}^{n} |S(i_a, i_h)|} \quad (7)$$

The SWA is an item-based prediction, which is the simple weighted average for user u_a of product i_a.The preference of each product is calculated for users. $S(i_a, i_h)$ is the similarity value between product i_a and i_h whereas r_{u_a,i_h} is the preference of user u_a and product i_h.

2.4 Related Works

When creating recommendation systems by applying collaborative filtering techniques, an important process is the rating data of users in the system. Problems regarding data sparsity and the volume of data dimension affects system creation, a few researchers have solved these problems by adjusting the technique to evaluate on the similarity value in various forms. For example, create the system by using fusing user and item information to deal with data sparsity using side information in the recommendation system (FUIR) [10]. This method applies the data dimension related to product quality in order to find the correlation by adapting the User Relative Interest (CTRI) technique to find the similarity with Cosine, which is the use of a data dimension of product qualities to determine the CTRI value between the different items. A New Likelihood-Based Similarity Score (LiRa) [11] uses the Likelihood technique in forms of a proportion to create an equation to find the similarity. Log10 is used to solve a problem regarding the similarity search of users who rank the preference at the same level, resulting in the level of the similarity = 1. Accordingly, this method is used to estimate the correlation by using the number of product co-rate value for consideration. Bhattacharyya [12] presented correlation search with the combination of different techniques. This first step is checking the create value based on the rank preference of products. The values are 0 and 1. If the value = 0, the Jaccard score will be applied as co-rate; whereas, if the value = 1, the consideration will apply the co-rate in the form of mean and median score and standard deviation in step. Proximity-Impact-Popularity (PIP) [5] divides information types on the basis of each case and manages 3 major activity data, namely Proximity, Impact and Popularity to calculate co-rate. In other words, this is an evolutionary approach to combine results of recommender systems techniques based on collaborative filtering [6]. The above applied a genetic algorithm (GA) using the genetic procedures to select results obtained from the co-rate of the combined techniques to develop recommendation systems. Each technique was applied to solve data sparsity problems, a genetic algorithm was used to select the sum which is the result obtained after discovering similarity of recommendation. Each technique was designed to solve data sparsity problems. Not only similarity improvement techniques have been used to develop recommendation systems but a few researchers have also applied clustering techniques [13,14]. Clustering techniques are considered as an important tool to expose qualification of data used for wide ranges of activities. The technique can also be applied to cluster data from their features. Clustering techniques that are mostly applied to large-scale data sets are hierarchical clustering, k-means and other clustering such as random sampling approach, randomized search approach, condensation-based approach, density-based approach and Grid-based approach. They are commonly applied to data with high dimensionality to increase the effectiveness of a clustering technique. Clustering techniques that are widely used to organize data also apply the simple k-means [15] when clustering due to its time-saving and ease. A few researchers have conducted experiments by applying clustering techniques with different types of data with the collaboration of other algorithms, such as the application of using k-means to cluster data in web pages that contain data sparsity which appear in large-scale [16] by turning k-means into a MINI-BATCH K-MEANS format, which results in effective time consumption when clustering data. To cluster data that contains a large scale of text, [17] research regarding concepts of decompositions for large sparse text data using clustering have applied k-means for clustering due to less time consumption; in other words, it is known as a technique to cluster binary data streams with k-means [18]. k-means clustering is mostly applied for fast data clustering. It is one of the easiest algorithms applied to divide the unsupervised learning group; it is a common solution when clustering problems. The k-means algorithm will divide the partition into k groups. Each group is represented by its average value which is used as group centroid to measure the distance of the data within the same group in the first step of clustering. To find k-mean, the number of group (k) requires specification and the centric requires the determination, started with k point(s); k represents the number. K-mean work processes are as follows: 1) Randomly select 'k' cluster centers. 2) Calculate the distance between each data point and cluster centers.

1) Randomly select 'k' cluster centers.

2) Calculate the distance between each data point and cluster centers.

3) Assign the data point to the cluster center whose distance from the cluster center is minimum of all the cluster centers.

4) Recalculate the new cluster center.

5) Recalculate the distance between each data point and new obtained cluster centers.

6) If no data point was reassigned then stop, otherwise repeat from step 3).

The advantages of k-means clustering are 1) When data is in a large-scale while the number of group is small, the application of k-means mean can be calculated faster than other clustering methods. 2) The process applied to find mean score using k-means might result in density of member within the group, especially if the group is in circle. The limitations are 1) It is not easy to find the appropriate k value. 2) When the data group is not in a circle, this technique might not work as well. And 3) there are limitations in terms of size, density and shape.

3. RESEARCH METHODOLOGY

Memory-based collaborative filtering technique was developed aiming to recommend items set in the case of data sparsity with simple approach and support to general system. Our developed method was divided into three parts. Part A; Simple Agent Module, was a part of the operation to find the agent of data set for creating the recommendation system. In this step, we tried to find a representative of the data sparsity to cluster for performance of recommendation system. Part B; Neighbor Filtering, was the process applied to find neighbors of the active users who just arrived in the system and to generate the data set for memory-based collaborative filtering technique by working, neighbors filtering from users-items matrix agent from experiment A. And Part C; Prediction and Recommend, was the process of predicting and recommending items for active users, users-items matrix from experiment B to recommend. This process used four similarity measurement to predicting items of each active user and to evaluate the error in the prediction. An example is illustrated below.

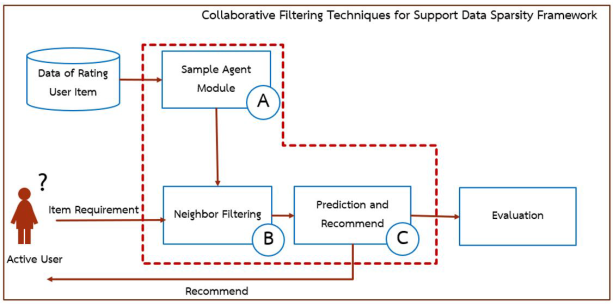


Fig.1: *Collaborative Filtering Technique to Support Data Sparsity Framework.*

Figure 1 shows the conceptual framework to create a recommendation system to support data sparsity. Main processes applied are: 1) Applying the data used to create the model to cluster and find the agent to create the recommendation system: Sample Agent Module (SAM) 2) Find the neighborhood 3) Generate predictions and recommendations and 4) Conduct the evaluation.

3.1 Sample Agent Module

The Sample Agent Module is a part of the operation to find the agent of data set to create the recommendation system. In this research, the data sets were created to evaluate 2 types of data namely Traditional Data (TD) and Statistic Data (SD). Clustering of each data set was also conducted in the experiment as illustrated below.

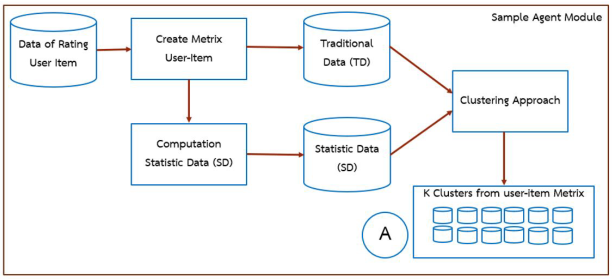


Fig.2: *Sample Agent Module.*

Figure 2 shows the Sample Agent Module process applied to data clustering. This applies an agent to create the recommendation system with an application to cluster using the k-means algorithm to find the average of the population to create groups. The author conducted a k-means clustering experiment with sample data sets which exposed data sparsity at different levels. The data set used in the experiment simulated the level of user satisfaction with 100 users and 50 products. The satisfaction level range of the product was 0-5. The data set contained different levels of sparsity, ranking from 0.16, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 0.95. The data value were randomized where the value of k = 6 from the total number of users. The experiments were conducted 100 times. The results are illustrated in the table below.

Table 1: *Result of Clustering 100 Rounds on Sparsity.*

Sparsity	G1	G2	G3	G4	G5	G6
0.16	12	13	22	14	21	18
0.20	12	21	18	12	22	15
0.30	23	10	15	18	21	13
0.40	18	19	12	15	14	22
0.50	12	22	14	20	17	15
0.60	23	15	11	11	25	15
0.70	25	13	11	24	18	9
0.80	11	34	10	10	11	24
0.90	7	57	8	3	19	6
0.95	11	5	9	7	61	7

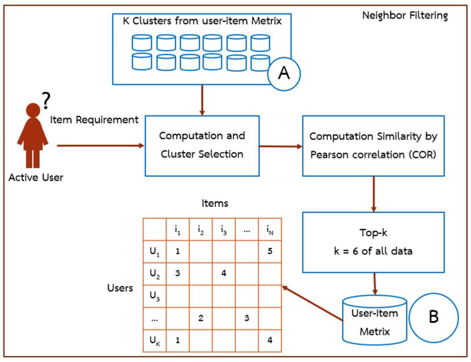
Table 2: *Result of Clustering 50 Rounds on Sparsity.*

Sparsity	G1	G2	G3	G4	G5	G6
0.16	18	15	19	10	16	22
0.20	14	20	11	16	17	22
0.30	13	23	17	17	17	13
0.40	19	16	17	15	14	19
0.50	12	16	27	18	12	15
0.60	16	20	14	14	23	13
0.70	25	13	11	24	18	9
0.80	23	9	17	22	22	7
0.90	26	15	9	14	2	34
0.95	11	8	15	10	51	5

Table 1 and Table 2 show the results of using the simple clustering technique. The k-means clustering was used in the experiment on data sparsity in different levels. The results showed that k-means clustering could work in system having data sparsity. And, it also worked normally in the system with normal data. Thus, simple cluster by k-means clustering technique was selected to cluster data sparsity in order to create the users-items matrix group for recommendation system in this study.

3.2 Neighbor Filtering

Neighbor filtering is the process applied to find neighbors of the active users who have just arrived in the system and to generate the data set to make a prediction of active users' preference. The processes are illustrated below.


Neighbor Filtering
K Clusters from user-item Metrix
A
?
Item Requirement
Active User
Computation and Cluster Selection
Computation Similarity by Pearson correlation (COR)
Top-k
k = 6 of all data
User-Item Metrix
B
Items
Users

	i_1	i_2	i_3	...	i_N
U_1	1				5
U_2	3		4		
U_3					
...		2		3	
U_K	1				4

Fig.3: *Neighbor Filtering.*

Figure 3 shows Neighbor Filtering is the application of the data set derived from the Sample Agent Module to find neighbors of active users. The initiation of the process appears when active users enter into the system, neighbors of the active users will be searched. The similarity or correlation between active users and agent group will also be evaluated with COR. The nearby neighbors are then selected to create data sets to make predictions of the preferences expressed by active users who enter the system after that.

3.3 Prediction and Recommendation

Prediction and Recommendation is the process of predicting the preference of active users entering the system. The prediction applies equation

$$Pu_a, i_a = \bar{r}_{u_a} + \frac{\sum_{h=1}^{n}(r_{u_h,i_a} - \bar{r}_{u_h}) \cdot S(u_a, u_h)}{\sum_{h=1}^{n} |S(u_a, u_h)|} \quad (8)$$

From Equation 8, it implies that the value derived from the prediction of i_a level exposed by the active users is based on active users' average score plus the sum of all products in the data set, the similarity value is then calculated. Thus, this research, applied the different similarity values to calculate the predictions and also compare the results as illustrated below.

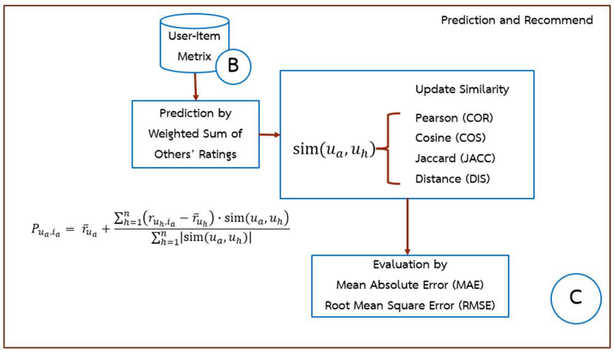

Prediction and Recommend
User-Item Metrix
B
Prediction by Weighted Sum of Others' Ratings
Update Similarity
$\text{sim}(u_a, u_h)$
Pearson (COR)
Cosine (COS)
Jaccard (JACC)
Distance (DIS)
$P_{u_a,i_a} = \bar{r}_{u_a} + \frac{\sum_{h=1}^{n}(r_{u_h,i_a} - \bar{r}_{u_h}) \cdot \text{sim}(u_a,u_h)}{\sum_{h=1}^{n}|\text{sim}(u_a,u_h)|}$
Evaluation by Mean Absolute Error (MAE) Root Mean Square Error (RMSE)
C

Fig.4: *Prediction and Recommend.*

Figure 4 shows Prediction and Recommendation, starting by using User-Item Metrix data from Top-k to find nearby neighbors. The prediction applies Sample Weighted Average, the similarity value used are namely Pearson Correlation, Cosine, Jaccard and Euclidean to predict and evaluate the effectiveness of the values derived from making predictions with the sample using Mean Absolute Error and Root Mean Square Error. Most exiting recommendation systems also applied the previous measurements.

3.4 Evaluation

Mean Absolute Error: MAE [1,3,19], is a measure of difference between an accurate value and an estimated value from a sample agent. If MAE is less, it defines that a sample agent could estimate an accurate value similar to the result of the test under an equation

$$MAE = \frac{1}{S}\sum\nolimits_{i,j} |R_{i,j} - \bar{R}_{i,j}| \tag{9}$$

$R_{i,j}$ is defined as a value obtained from a prediction of a sample agent. $\bar{R}_{i,j}$ is thereby an actual value while S is a number of the data used in a sample agent.

Root Mean Square Error: RMSE is a measure of a discrepancy similar to square root of a standard deviation. It is found to have less value when representing a sample agent and a low value to an actual value as in the below equation

$$RMSE = \sqrt{\frac{1}{S}\sum\nolimits_{i,j}(R_{i,j} - \bar{R}_{i,j})^2} \tag{10}$$

To define $R_{i,j}$, it is a predicted value from a sample agent. $\bar{R}_{i,j}$ is an accurate value, and S is a number of the data in a sample agent.

Both MAE and RMSE are the measure of discrepancy from both of the averages based on a calculation. If it equals 0, it represents the best value because no discrepancy occurred from the calculation. RMSE results in a higher value than MAE due to the fact that the discrepancy is squared. If considering large discrepancy or huge mistake, RMSE has higher possibility to provide false results.

3.5 Recommendation for Users

The research about the prediction on the preference rank and recommendation were conducted by using traditional data and statistic data to find a suitable data set or similarity for a new system of preference rank prediction and user recommendation. From researching user recommendation results of prediction and recommendation were obtained. A previous research found the first three types of products a buyer preferred. The data set from grouping was then used for prediction classifying into 4 types as follows:

1) Traditional Data (TD): a group of the data from the total users' preference rank used for grouping.

2) Statistic Data (SD): a group of the data from the total users' preference rank used for grouping.

3) Traditional Data Union Statistic Data (TD ∪ SD): Aggregating new group of the data from TD group with SD group.

4) Traditional Data Intersect Statistic Data (TD ∩ SD): Forming a new group of the data formed by TD group and SD group.

From these 4 earlier mentioned types, population was screened, and a list of products was introduced to active users in the system via the following procedure.

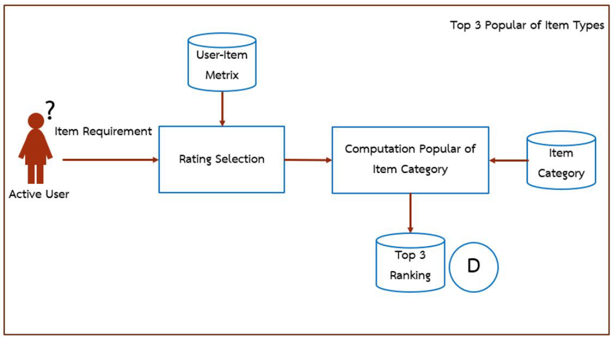


Fig.5: *The preference ranking of item types.*

Figure 5 shows the preference rank of item types from active users which was calculated by each new user's preference ranks before measuring the interests of the active users to item types via their first three preference ranks. This was to compare and analyze the prediction item list to users, as shown in the figure 6.

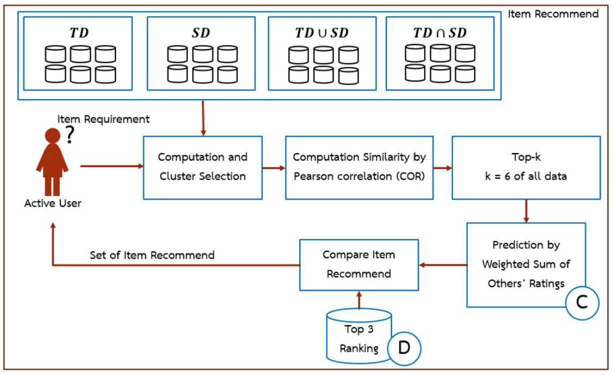


Fig.6: *Prediction of item list to active users in the system.*

Figure 6 shows the prediction of item list to active users in the system was conducted by using all 4 data types to screen population as a result of users' item preference rank.

4. RESULT AND DISCUSSION

4.1 Result of Scenarios

4.1.1 Data Preparations

From the research methodology, movie lens dataset (1M) consisting of 671 users and 163,949 items were used, this shows item appreciation range (1,5), and the users' satisfaction (98,903 in total) as shown below.

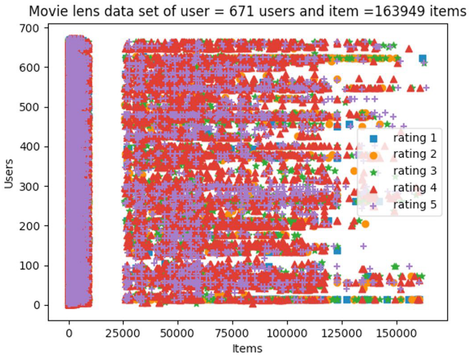


Fig.7: *Scatter Plot of Movie lens Dataset.*

Figure 7 shows the level of the users' satisfaction towards a list of times equivalent to 1-5 with each level as follows.

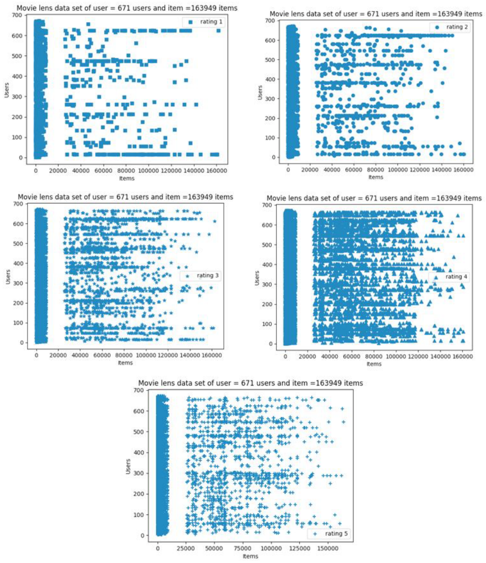


Fig.8: *Scatter Plot of Movie lens Dataset Rating 1 to 5.*

Figure 8 shows the characteristic of the data in each level of satisfaction towards items ranging from Rating 1, Rating 2, Rating 3, Rating 4, and Rating 5 in the movie lens dataset.

In the research, data selection was used to train and test model of sparsity for creating a recommendation system. This affected the efficiency of recommendation system building. The data in our research was divided into two sets: 90% and 10%. The research divided the data into two sets: 90% and 10% of the users that included training data to build the model, and test data as an active user for the research experiment. Training Data was clustered classifying two types: Traditional Data and Statistic Data.

Traditional Data was built based on the level of users' satisfaction totaling 604 users and 163,949 items. The data was arranged in Metrix (60416×3,949).

Statistic Data was obtained from the statistical Traditional Data through data clustering. The data consisted of Count All Ratings, Average, Mode, Standard Deviation, Count Rating 1, Count Rating 2, Count Rating 3, Count Rating 4, Count Rating 5, and Count Rating 0 as shown in the table 3.

Table 3: *Description of Statistic Data.*

Attributes	Description
Count	Count all rating of TD
Average	Average rating each user
Mode	Mode rating each user
SD	Standard Deviation rating each user
R1	Count rating 1 each user
R2	Count rating 2 each user
R3	Count rating 3 each user
R4	Count rating 4 each user
R5	Count rating 5 each user
R0	Count not rating each user

Table 3 shows the description of attributes and the following SD data were used in the clustering.

4.1.2 Defining Sample Agent

From the research based on two sets of data, K-means clustering defined k equivalent to 0.01 of the 6 user groups in 20 rounds of the calculations. The users were clustered as follows.

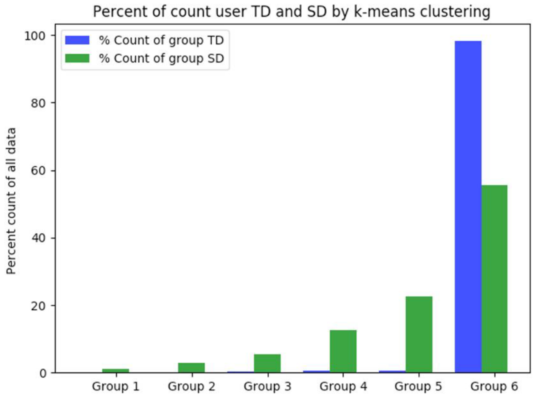


Fig.9: *Percent of Count User TD and SD by k-means.*

Figure 9 shows the percentage of the number of members in each group including the k-means clustered data set: TS and SD representing a neighbor search and rating selection for the next prediction.

4.1.3 Item Recommending

To select neighbor, the data of Active Users were used to search representatives based on the below equation

$$sim(u_a,u_b)=\frac{\sum_{h=1}^{n}(r_{u_a,i_h}-\bar{r}_{u_a})(r_{u_b,i_h}-\bar{r}_{u_b})}{\sqrt{\sum_{h=1}^{n}(r_{u_a,i_h}-\bar{r}_{u_a})^2}\sqrt{\sum_{h=1}^{n}(r_{u_b,i_h}-\bar{r}_{u_b})^2}}$$

From the equation of similarity, users in a sample agent had similarity with Active users accessing the system. Then, it proceeded Top-k; k=6 for rating prediction of 67 users from the equation of Weighted Sum of Others' Ratings

$$Pu_a,i_a=\bar{r}_{u_a}+\frac{\sum_{h=1}^{n}(r_{u_h,i_a}-\bar{r}_{u_h})\cdot S(u_a,u_h)}{\sum_{h=1}^{n}|S(u_a,u_h)|}$$

The rating prediction was calculated by similarity by defining

$$S(u_a,u_h)=[COR(u_a,u_h),Cosine(u_a,u_h),JACC(u_a,u_h),DIS(u_a,u_h)]$$

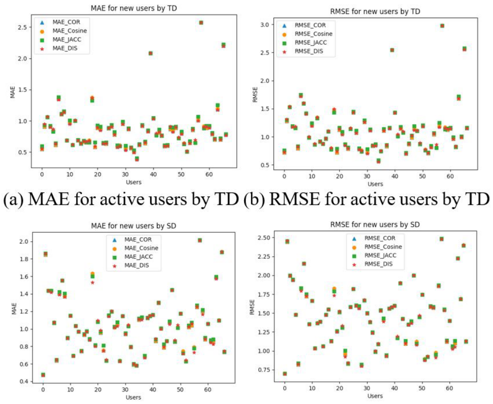
Table 4: *Description of Similarity.*

Similarity	Description
$COR(u_a,u_h)$	Similarity by Pearson correlation user u_a and u_h
$Cosine(u_a,u_h)$	Similarity by Cosine user u_a and u_h
$JACC(u_a,u_h)$,	Similarity by Jaccard user u_a and u_h
$DIS(u_a,u_h)]$	Similarity by Euclidean user u_a and u_h

Table 4 shows similarity obtained from the various calculation to predict the rating of each active user.

4.1.4 Evaluation of Recommendation System

The experiment focused on an accurate value of active users from Mean Absolute Error: MAE and Root Mean Square Error: RMSE.



(a) MAE for active users by TD (b) RMSE for active users by TD

(c) MAE for active users by SD (d) RMSE for active users by SD

Fig.10: *MAE and RMSE of TD and SD.*

Figure 10 shows MAE and RMSE data sets, TD and SD represent error from prediction of the rating of Active users. The average of all Active users are showed in the following table.

Table 5: *Mean Absolute Error: MAE.*

	Traditional Data(TD)	Statistics Data(SD)	|TD-SD|
MAE_COR	0.857178322	1.043255773	0.18607745
MAE_Cosine	0.857225429	1.043223156	0.18599773
MAE_JACC	0.872280905	1.049827599	0.17754669
MAE_DIS	0.858297143	1.040316747	0.18201960

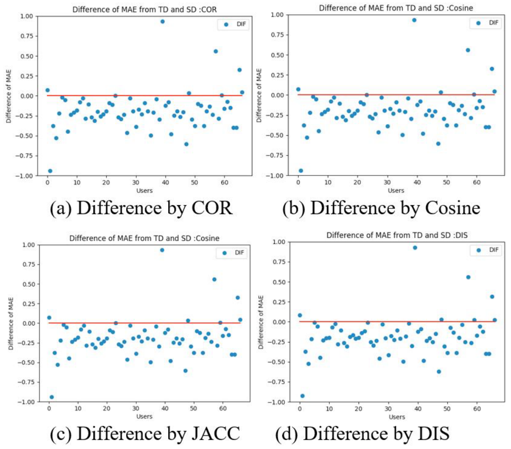
Table 6: *Root Mean Square Error: RMSE.*

	Traditional Data(TD)	Statistics Data(SD)	|TD-SD|
RMSE_COR	1.118590972	1.451273272	0.33268230
RMSE_Cosine	1.118606444	1.451237410	0.33263097
RMSE_JACC	1.130071765	1.455747542	0.32567578
RMSE_DIS	1.120553540	1.446934450	0.32638091

Table 5 and Table 6 present the comparison of MAE of each similarity measurements in predicting item rating by comparing error values from TD and SD data sets. TD is traditional data which general recommendation system use in creating model. And, SD is traditional statistic data information which being created by the research. In the research, accuracy was evaluated by considering on error of the similarity measurements in each prediction rating of items.

4.2 Similarity Measurement Discussion

From the experiment of creating a recommendation system by selecting sample agents using k-means clustering of 2 groups of data, which are TD and SD, by predicting various similarities, allowed the author to show error data from the prediction via similarity as follows.



(a) Difference by COR (b) Difference by Cosine

(c) Difference by JACC (d) Difference by DIS

Fig.11: *Difference of MAE from TD and SD by COR, Cosine, JACC and DIS.*

Figure 11 shows mean absolute error received from using TD and TD data predicted by DIS, which resulted in MAE value of TD less than SD, 60 users and

SD less than 7 users. The MAE value resulted from each similarity measurement was an average of the error of all user. After considering the number of users with lower error values in TD and SD, it showed that the error in number of users using the similarity with Euclidean of TD was less than that of using the other similarity in TD and SD data set. Thus, we selected Euclidean similarity for recommending items.

From the research process, the author proceed to create a recommendation system and separated the data into 2 groups, which are TD and SD. SD data is the data received from finding the highest statistic from TD data. This will be used as sample agents of TD when clustering to find the sample agent. Once completed, the recommendation system from both data groups using similarity in COR, Cosine, JACC, and DIS was created to calculate the satisfaction of the users and test with MAE and RMSE values of both groups of data. From the experiment, the MAE of TD data is less than SD to build the model, but when considering each active user shown in Figure 11. One can see that the use of DIS similarity could yield the best rating prediction of TD data as MAE in each active user from rating prediction of DIS in TD data has lower value than SD data.

4.3 Results of Recommendation for Users

We offered recommendation or showing some items set for active user's preference. In the research we considered user preference and recommend the items set in that type to the active user. Then, three users were randomized to test the recommend Items set for active user' preference in data sparsity by using Euclidean (DIS) prediction was measured as a testing result. The TD data set could be predicted by its preference ranks. In this process, all data of the items were deliberated with its recommendation to active users. Moreover, all the data sets were ranked based on the users' preferences towards items in each scale as shown in Table 7.

Table 7: *Total Rating of Item Type.*

No	Item Type	Number of Rating
1	Action	1,545
2	Adventure	1,117
3	Animation	447
4	Children	583
5	Comedy	3,315
6	Crime	1,100
7	Documentary	495
8	Drama	4,365
9	Fantasy	654
10	Film	133
11	Horror	877
12	Musical	394
13	Mystery	543
14	Romance	1,545
15	Sci	792
16	Thriller	1,729
17	War	367
18	Western	168
19	(no genres listed)	18

Table 7 shows the three-scale rank of the item preferences classified by each type given by users. The first rank were Drama, Comedy, and Thriller, respectively. Then, three users were randomized to test the recommendation of the requested items and each user has a different preference for each item type: User 1 is less, User 2 is moderate and User 3 is high given rating items. The result shows the preference rank of all three users are shown below.

Table 8: *Total Rating of 3 Users.*

No	Type	User 1	User 2	User 3
1	Action	8	180	190
2	Adventure	5	117	180
3	Animation	2	13	48
4	Children	2	21	91
5	Comedy	3	255	798
6	Crime	19	165	340
7	Documentary	0	21	173
8	Drama	16	453	1478
9	Fantasy	1	54	114
10	Film	14	11	91
11	Horror	0	39	115
12	Musical	0	11	173
13	Mystery	11	61	179
14	Romance	3	125	546
15	Sci	2	79	122
16	Thriller	19	204	402
17	War	1	65	73
18	Western	0	16	40
19	(no genres listed)	0	0	2

Table 8 shows the preference rank of each user who gave his/her item type preference rank. The items that users put in the top 3 were used to test the user recommendation system as follows. User no. 1 arranged his/her preferences into the top 3: Crime, Thriller, and Drama, in ascending order. User no. 2 ascendingly ordered the top 3 preferences: Drama, Comedy, and Thriller. Lastly, user no.3 showed his/her ascending preference orders in the top 3: Drama, Comedy, and Romance. To proceed prediction of preference ranks between users and items, all data sets were used to screen the population for a recommendation. All the above data sets were used for item preference ranks with 3 active users in the system. Data of each product were recommended to users through prediction of the top 3 of the item list as shown in the following table.

Table 9: *Item Recommended for Users by SD.*

No	Item	Type	Correct
1	4007	Drama	Y
	26151	Crime|Drama	Y
	36527	Drama	Y
2	56782	Drama|Western	Y
	2351	Drama	Y
	1860	Drama	Y
3	67504	Documentary	N
	83411	Comedy	Y
	83359	Comedy	Y

Table 9 shows items were recommended on the system to each new user by calculating the SD data set in order to predict preference rank of users.

Table 10: *Item Recommended for Users by TD.*

No	Item	Type	Correct
1	26151	Crime|Drama	Y
	3920	Drama|Fantasy|Mystery|Romance	Y
	3966	Crime|Film-Noir	Y
2	2572	Comedy|Romance	Y
	2947	Action|Adventure|Thriller	Y
	5071	Drama|Romance	Y
3	1111	Documentary	N
	306	Drama	Y
	6299	Documentary	N

Table 10 shows items were recommended on the system to each new user by calculating the TD data set in order to predict preference rank of users.

Table 11: *Item Recommended for Users by TD∪SD*

No	Item	Type	Correct
1	26151	Crime|Drama	Y
	3920	Drama|Fantasy|Mystery|Romance	Y
	3966	Crime|Film-Noir	Y
2	6918	Drama	Y
	7136	Comedy|Drama|Romance	Y
	2068	Drama|Fantasy|Mystery	Y
3	67504	Documentary	N
	83411	Comedy	Y
	83359	Comedy	Y

Table 11 shows items were recommended on the system to each new user by calculating the TD∪SD data set in order to predict preference rank of users.

Table 12: *Item Recommended for Users by TD∩SD*

No	Item	Type	Correct
1	4007	Drama	Y
	26151	Crime|Drama	Y
	36527	Drama	Y
2	Null	Null	N
3	1111	Documentary	N
	306	Drama	Y
	6299	Documentary	N

Table 12 shows items were recommended on the system to each new user by calculating the TD∩SD data set in order to predict preference rank of users. Table 9 to Table 12 shows the preferences of the items based on active users in the system. The prediction on the item list preference rank of the item list was initiated to recommend active users in the system. This was shown in a form of the top 3 preference within more 3 item lists. The item list number and the items matched with the user's top 3 preferred item type based on all 4 data sets shown in the following table.

Table 13: *Item Recommended Top 3 All Data Sets*

Data Sets	User 1	User 2	User 3
Only SD	3	3	2
Only TD	3	3	1
TD∪SD	3	3	2
TD∩SD	3	0	1

Table 13 shows is a summary table of recommended items for 3 users that match the type of item that the user would prefer. This research used the item list in the system to recommend active users using 4 data sets with item preference ranks. The item lists and the categories of the items could be shown covering the top 3 to active users when recommended and predicted as shown in Table 14.

Table 14: *Item Recommended for Users by All Data Sets*

Data Sets	**User 1**	**User 2**	**User 3**
Only SD	6	9	4
Correct	6	9	3
Only TD	3	44	8
Correct	3	42	4
TD∪SD	3	4	4
Correct	3	4	3
TD∩SD	6		8
Correct	6		4

Table 14 shows the item lists in the recommendation system processing each new user in the system. The required SD data set recommends the item lists ranging from 6, 9, and 4 to users no.1-3. The TD data set recommended item lists ranging from 3, 44, and 8 to users no. 1-3 being. While, TD∪SD data set recommended item lists ranging from 3, 4, and 4 to users no. 1-3. TD∩SD data set recommended item lists ranging from 6 and 4 to user no. 1-3. The ratio of item lists matching with item types which users preferred are shown in Table no. 15.

Table 15: *Ratio of Item Recommended*

Data Sets	User 1	User 2	User 3
Only SD	1	1	0.75
Only TD	1	0.95455	0.5
TD∪SD	1	1	0.75
TD∩SD	1		0.5

Table 15 shows the item lists that were recommended by users to 3 users. The ratio of this is shown as follows.

1)Statistic Data (SD) had a ratio of 1st user to 1 equivalent to 6 recommended items while 2nd user to

1 equalled to 9 recommended items, and 3rd user to 0.75 equivalent to 9 recommended items.

2)Traditional Data (TD) had a ratio as 1st user to 1 equivalent to 3 recommended items while 2nd to 0.955455 equalled to 44 recommended items, and 3rd to 0.5 was equalled to 8 recommended items.

3)Traditional Data Union Statistic Data ($TD \cup SD$) had a ratio of 1st to user 1 equivalent to 3 recommended items while 2nd user to 1 equalled to 4 recommended items, and 3rd user to 0.75 equalled to 4 recommended items.

4)Traditional Data Intersect Statistic Data ($TD \cap SD$) had a ratio of 1st user to 1 equivalent to 6 recommended items, and 3rd user to 0.5 equalled to 8 recommended items.

From the research methodology, a set of item lists presented to active users in the system can be further explained as follows.

Table 16: *Item Recommended Sets*

Data Sets	Item Recommended Sets
Only SD	$AU_1 = \{i_1, i_2, i_3, \ldots, i_6\}$ $AU_2 = \{i_1, i_2, i_3, \ldots, i_9\}$ $AU_3 = \{i_1, i_2, i_3, i_4\}$
Only TD	$AU_1 = \{i_1, i_2, i_3\}$ $AU_2 = \{i_1, i_2, i_3, \ldots, i_4\}$ $AU_3 = \{i_1, i_2, i_3, \ldots, i_8\}$
TD∪SD	$AU_1 = \{i_1, i_2, i_3\}$ $AU_2 = \{i_1, i_2, i_3, i_4\}$ $AU_3 = \{i_1, i_2, i_3, i_4\}$
TD∩SD	$AU_1 = \{i_1, i_2, i_3, \ldots, i_6\}$ $AU_3 = \{i_1, i_2, i_3, \ldots, i_8\}$

Table 16 shows item lists in the recommendation system used by all 4 types of the data set to give users predictions and recommendations. AU represented each new user in which 1st user showed the accurate list of item recommendation from the system of all data sets classified by the users' preferred item types followed by 2nd user and 3rd user. The number of item types in the system were recommended in the data $Only\ TD$ as the most highly recommended item list; therefore, Euclidean (DIS) was used to find the nearest value in TD for the item list recommendation to active users in the system with the highest number of recommendations.

5. CONCLUSIONS AND FUTURE WORK

For the creation of memory-based collaborative filtering techniques in a recommendation system, the most important area is the number of user's rating in the system as this affects the calculations. If the data is light and its dimension is large, it will affect the performance of recommendation system. We offer recommend or show some items set for user's preference. This research presented the Top-k Items Recommendation System and have methods was divided into three parts included 1) Simple Agent Module 2) Neighbour Filtering and 3) Prediction and Recommend, which uses the process of finding a sample agent from data sparsity, then applies creation of filtering collaborated with k-means clustering to separate data into 2 groups, which are TD and SD. From the test of k-means clustering in data groups with different sparsity, it was found that k-means clustering is compatible with different data sparsity to create sample agents in groups for the creation of a recommendation system and can also use data from clustering as the sample agent in the creation of a recommendation system. The process of creating a recommendation system works by finding nearby neighbors. In the creation and recommendation of the users, the author used COR to find nearby neighbors by setting Top-k of the data group in the prediction with satisfaction at 0.5, quality with Weighted Sum of Others' Ratings (WS) and change similarity by COR, Cosine, JACC, and DIS to find the satisfaction of active user and test the performance of satisfaction prediction with MAE and RMSE. This resulted in the understanding that using DIS similarity can rate a prediction with the consideration of MAE in each data group. From results of this research, a recommendation system was created using WS prediction base on the Euclidean similarity with 4 data sets: Only TD, Only SD, TD∪SD and TD∩SD. These were based on user's preference and were used to find the value in TD from the item list. This provided recommends to active users in the system referring to the highest number of recommended items in the list.

The contribution of this research is that it could be used as a fundamental base when using clustering to increase the performance of a sample agent and also improve the similarity and rating of predictions to further improve recommendation systems.

References

[1] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutirrez, "Recommender systems survey," *Knowledge.-Based Syst.*, vol. 46, pp. 109-132, Jul. 2013.

[2] H. Ma, I. King, and M. R. Lyu, "Effective Missing Data Prediction for Collaborative Filtering," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New York, NY, USA, 2007, pp. 39-46.

[3] X. Su and T. M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," *Adv Artif Intell*, vol. 2009, pp. 4:2-4:2, Jan. 2009.

[4] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, "Recommender system application developments: A survey," *Decis. Support Syst.*, vol. 74, pp. 12-32, Jun. 2015.

[5] H. J. Ahn, "A new similarity measure for collaborative filtering to alleviate the new user cold-

starting problem," *Inf. Sci.*, vol. 178, no. 1, pp. 37-51, Jan. 2008.

[6] E. Q. da Silva, C. G. Camilo-Junior, L. M. L. Pascoal, and T. C. Rosa, "An evolutionary approach for combining results of recommender systems techniques based on collaborative filtering," *Expert Syst. Appl.*, vol. 53, pp. 204-218, Jul. 2016.

[7] Shimodaira, Hiroshi, "Similarity and recommender system," *Sch. Inform. Univ. Eidenburgh 21*, no. 2014.

[8] Thanaphon Pukseng and Sunantha Sodsee, "Applying Expert Concept for Recommendation System," *Journal of Science and Technology*, vol. 25, no. 2, pp. 361-375, 2017.

[9] Tatiya, R. V. and P. A. S. Vaidya, "A Survey of Recommendation Algorithms," *IOSR J. Comput. Eng.*, vol. 16(6), pp. 16-19, 2014.

[10] J. Niu, L. Wang, X. Liu, and S. Yu, "FUIR: Fusing user and item information to deal with data sparsity by using side information in recommendation systems," *J. Netw. Comput. Appl.*, vol. 70, pp. 41-50, Jul. 2016.

[11] V. Strnadova-Neeley, A. Buluc, J. R. Gilbert, L. Oliker, and W. Ouyang, "LiRa: A New Likelihood-Based Similarity Score for Collaborative Filtering," ArXiv160808646 Cs, Aug. 2016.

[12] B. K. Patra, R. Launonen, V. Ollikainen, and S. Nandi, "A new similarity measure using Bhattacharyya coefficient for collaborative filtering in sparse data," *Knowledge.-Based Syst.*, vol. 82, pp. 163-177, Jul. 2015.

[13] R. Xu and D. Wunsch, "Survey of clustering algorithms," *IEEE Trans. Neural Netw.*, vol. 16, no. 3, pp. 645-678, May 2005.

[14] W. Niyagas, A. Srivihok, and S. Kitisin, "Clustering e-Banking Customer using Data Mining and Marketing Segmentation," *ECTI Trans. Comput. Inf. Technol. ECTI-CIT*, vol. 2, no. 1, pp. 63-69, 2006.

[15] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognit. Lett.*, vol. 31, no. 8, pp. 651-666, Jun. 2010.

[16] D. Sculley, "Web-scale K-means Clustering," in *Proceedings of the 19th International Conference on World Wide Web*, New York, NY, USA, 2010, pp. 1177-1178.

[17] I. S. Dhillon and D. S. Modha, "Concept Decompositions for Large Sparse Text Data Using Clustering," *Mach. Learn.*, vol. 42, no. 1-2, pp. 143-175, Jan. 2001.

[18] C. Ordonez, "Clustering Binary Data Streams with K-means," in *Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, New York, NY, USA, 2003, pp. 12-19.

[19] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating Collaborative Filtering Recommender Systems," *ACM Trans Inf Syst.*, vol. 22, no. 1, pp. 5-53, Jan. 2004.

Kittisak Onuean received the B.Tech. degree in Business Computer from Burapha University, Thailand in 2003, and M.Sc. degree in Information Technology, Burapha University, Thailand in 2010. He received Ph.D. degree in Information Technology, King Mongkut's University of Technology North Bangkok, Thailand in 2018. He is currently a lecturer at Faculty of Science and Social Sciences, Burapha University, Thailand. His current research works concern recommendation systems, data analytics, and software development.

Sunantha Sodsee received the B.Eng. degree in Telecommunication Engineering from King Mongkut's Institute of Technology Ladkrabang (KMITL), Thailand in 2002, M.Sc. degree in Information Technology (International Program) from King Mongkut's University of Technology North Bangkok, Thailand in 2005, and Dr.-Ing. degree in Chair of Communication Network form FernUniversitt in Hagen, Germany in 2011. She received Ph.D. degree in Information Technology (International Program), King Mongkut's University of Technology North Bangkok, Thailand in 2012. She is currently a lecturer at Faculty of Information Technology King Mongkut's University of Technology North Bangkok, Thailand. She is an Assistant Professor in Information Technology. Her current research interests are Artificial Intelligence, Evolutionary Algorithms, Data Mining, Complex Network Routing, P2P Networks, Data Communication, Recommender Systems, Decision Support Systems, Social Network Analysis, Multi-agent Systems and Consensus Problems.

Phayung Meesad received the B.Sc. degree in Technical Education (Teacher Training in Electrical Engineering) from King Mongkut's University of Technology North Bangkok, Thailand in 1994, M.Sc. degree in Electrical Engineering from Oklahoma State University, USA. in 1998, and Ph.D. degree in Electrical Engineering from Oklahoma State University, USA. in 2002. He is currently a lecturer at Faculty of Information Technology King Mongkut's University of Technology North Bangkok, Thailand. He is an Associate Professor in Information Technology, Faculty of Information Technology King Mongkut's University of Technology North Bangkok, Thailand. His current research works concern Computational Intelligence, Artificial Intelligence, Machine Learning, Deep Learning, Data Analytics, Data Science, Data Mining, Digital Signal Processing, Image Processing, Business Intelligence, Time Serires Analysis and Natural Language Processing.