Real Estate Recommendation Using Historical Data and Surrounding Environments

Uchchash Barua, Md. Sabir Hossain, Mohammad Shamsul Arefin[*]
Department of Computer Science & Engineering
Chittagong University of Engineering & Technology, Raozan, Chittagong.
Email: shrabon56@gmail.com, {sabir.cse, sarefin}@cuet.ac.bd
*Corresponding Author

Abstract—Recommending appropriate things to the user by analyzing available data is becoming popular day by day. There are no sufficient researches on Real-estate recommendation with historical data and surrounding environments. We have collected real-estate, historical and point of interest (POI) data from the various sources. In this research, a hybrid filtering technique is used for recommending real-estate consisting of collaborative and content-based filtering. Generally, in every website user ratings are collected for the recommendation. But we have considered historical data and surrounding environments of a real-estate location for recommendation by which it will be easy for a user to decide that which place would be better for him/her. If any user request for any specific location then the system will find the POI data using google map API. Then the system will consider historical data of that area, got from the trusted sources. So considering the minimum price and optimal facilities, our system will recommend top-k real-estate. After extensive experiments on real and synthetic data, we have proved the efficiency of our proposed recommender system.

Index Terms—Real-estate, Point of interest, Top-k apartments, recommendation, collaborative filtering.

I. INTRODUCTION

Online Oxford Dictionary (http://www.oed.com/) defined, "Real estate is the property consisting of land and the buildings on it, along with its natural resources such as crops, minerals or water; the immovable property of this nature and its surrounding environments like hotels, hospitals, schools etc.". In the wide term, Real estate management is the control and oversight of real estate. Real estate management manages the necessaries required including personal property, machinery, instruments, and physical capital assets. The amount of potentially interesting content is increasing day by day. A real estate recommender system can help people cope with the situation. It becomes a nightmare to go through an enormous amount of available potential content to select the best ones. Besides, since the price and demand for renting or buying houses or lands are rising higher now-a-days, the real-estate market is becoming popular day by day all over the world.

Statistics exhibits that Dhaka, capital of Bangladesh, is the high market for housing finance. Recent reviews have pointed out that over 70% of the homes are rented in Dhaka and there is an annual requirement of over 60,000 new residences [1]. Bangladesh is still experiencing a vulnerable economic system that hinders financial savings and investments, for this reason, financial development.

The motivation behind the research is to show how a recommender system can be applied in Real estate scenario. The tremendous pressure on land may and the willingness for settlement may provide with many choices; on the other hand, users may sometimes feel confused about a wise investment of their wealth and about to be benefitted. So, while designing the recommender system we have considered, 1) the existing problems in our country associated with real estate, 2) all the information available about a real-estate for user satisfaction and, 3) the right content to the right person

The growth rates of urban have increased around 37% over the last 7 years where the growth rates in real estate apartments are 9%, reported in [**1**]. The stack holders of real estate are always in uncertainty where to be deposited. Here, we have chosen a web-based real estate recommender system mainly for the people of Bangladesh that not only deals with the present situation of the country but also provides the great user experience.

The main contributions of this research can be summarized as follow:

- To analyze historical data of the real estate in Bangladesh
- To evaluate the surrounding environments of the requested location
- To recommend the sectors of real estate based on historical data and user criteria.

II. RELATED WORK

There has been a lot of researches on content-based and collaborative filtering technique for constructing a recommender system.

Atisha Sachan et al. [2] granted a survey on recommender system based totally on collaborative filtering where they discuss the method to find out the interest of the user. They designed a book recommender system considering four sorts of filtering approach particularly demographic technique, content-based filtering, collaborative filtering, and hybrid method. They discussed stability and cold-start problem. But they didn't consider sparsity issue in details. Another important work by Prem Melville et al [3] where they defined different types of recommender system with associated limitations and challenges. Jyoti Gupta and Jayant Gadge analyzed the performance of recommendation systems based on content-based filtering and demographics [4]. They solved the cold-start problem associated with the recommender system.

Feng et al. proposed an ontology-based InfoSlim architecture [5] to recommend relevant news to the users. They used lexical-level cosine and semantic-level ontology-based method to measure the similarity of news. They didn't discuss how to deal with the massive volume of information considering the history of reading, the profile of user and item of news for a specific device.

A Book Recommendation System (BRS) using a hybrid algorithm was proposed by Mathew et al. [6]. They used the combination of content-based filtering, collaborative filtering, and association rule mining technique to generate more accurate and relevant results. They didn't evaluate their proposed system in any specific metric. A similar work to recommend quality books to the user was proposed by Anand et al. [7]. They also used content-based, collaborative filtering and association rule mining algorithm to generate the recommendation.

In Paper [8], Chhavi Rana and Sanjay Kumar Jain Proposed a new book recommendation system considering the user's similarity and the choice of users changes over time. They have tested the performance of their proposed method empirically. They didn't consider contextual information and multiple rating to generate more accurate recommendations.

Zhichao et al. presented a hotel recommendation system [9] considering the surroundings. They measured the surrounding environments, user's reviews and calculate user preferences. They selected top-k hotels based on the similarity score between hotels and user preferences. But they didn't consider check-in trajectories of larger datasets.

Ding et al. [10] summarized the location-based social network recommender system to focus the objects and objectives by considering 3D social relationships. They indented to work on the trip recommendation to a group and sequence of activities of a user or group.

Vincent et al. presented a novel work using historic GPS data for collaborative location and activity recommendation [11]. They developed a system that could recommend the places based on the previously visited places by a user. They also recommend the related facilities associated with that places. They used user's location and activity histories as input and applied collective matrix factorization method to generate such significant results. They didn't consider personalized recommendation based on user preferences and activities.

Another location-based recommendations using Skyline query was presented by Kodama et al. in [12]. They showed the skyline objects together which is not user-friendly. They didn't consider dominance relationship as well as preference information.

As the classical collaborative filtering algorithms have data sparsity and high time complexity problem, Wang et al. [13] proposed a cloud-based multidimensional approach to recommend for new locations.
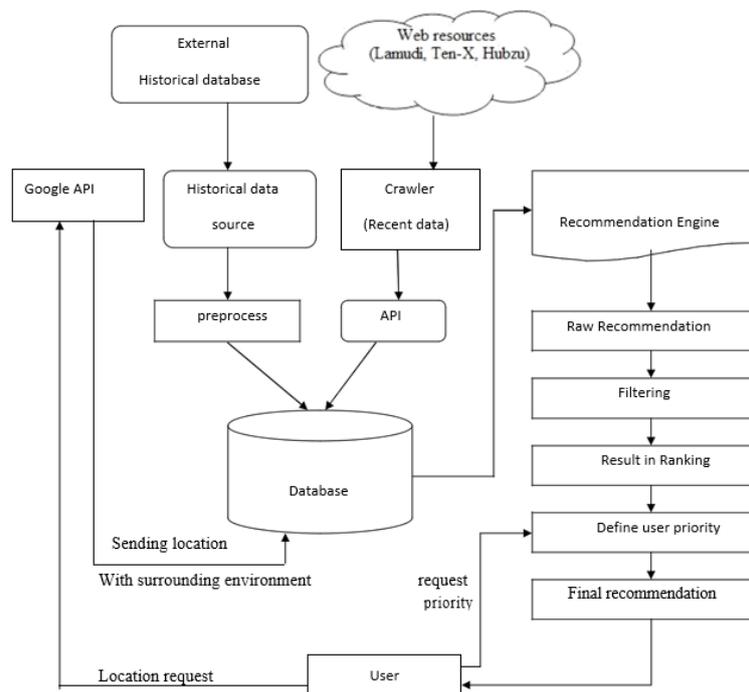

Fig.1. System Architecture for Real-estate Recommender System

III. DEVELOPMENT OF PROPOSED RECOMMENDER SYSTEM

The detailed architecture of the Real-estate Recommender System is provided in this section. There are mainly four sections: 1) The system architecture and flow chart of our system where different modules of the architecture and relationship among them are described briefly, 2) The storage module, 3) The analytical representation of our system which gives the details of the developed system with different algorithms and tables required implementation, and 4) The complexity analysis of our proposed algorithms.

A. Proposed System Architecture

Overview of the proposed Real-estate recommender system is shown in Fig. 1. We have sub-divided the whole system into four modules. Firstly, the user has to register an account and then log in. We store user's information on the database. We have collected the contents of Real-estates from various APIs and other sources like Google map API and LAMUDI's API.

We have collected ratings from about a hundred users and stored them in JSON format. We have used the Euclidean Distance method for finding similar users and items. For recommendation, we have used collaborative filtering. We have generated a weighted score to rank the apartments for the users. After normalizing and ranking we make top-N recommendations to each user. Flowchart of the working procedure of the recommender system is given in Fig. 2.
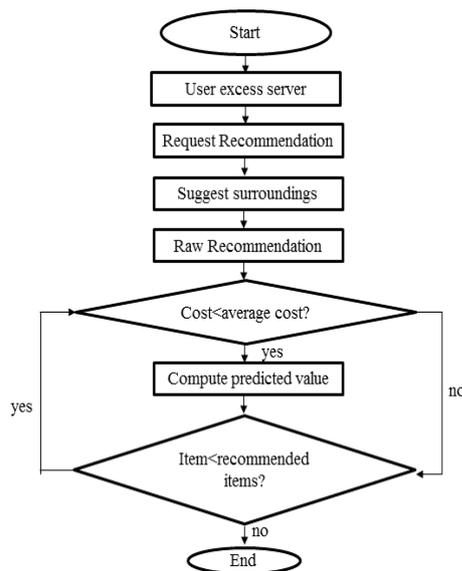

Fig.2. Flow Graph of our System

B. Storage Module

The user's information including register, login will store in the database. Also when they rate the items, the ratings will be stored in the database. For further uses, we've stored real-estate information along with latitude and longitude in the database. We've stored point of interest JSON information for every apartment in the database. We got that information from google map API. We've also stored the historical data of places in Bangladesh.

C. The Analytical Module

In this section, we explain in detail about the working procedure of our proposed real-estate recommender system. This section is subdivided into several subsections namely, 01) Finding similarity among the users, 02) Analyzing point of interest database, 3) Forecasting the historical data, 04) Predicting approximate rating of a user and, 05) Generate top-N apartments to recommend for a specific user.

1) Finding Similar user

User-based collaborative filtering technique is used to generate recommendations related to similar users. To find the similarity among the users we have used Euclidean Distance method. Euclidean distance method can be referred to the time needed to go to one point from another point. It forms coordinates to put preference values between items and measures Euclidean distance between each point. When distance value between two points, $sim\ (i,j)$ is large, it means the two points are not similar [16]. When $sim\ (i,j)$ is small, it means two points are similar.

The Euclidean distance formula for two item is given below.

$$r_2(\text{x,y}) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} = \sqrt{\sum_{i=1}^{2}(x_i - y_i)^2} \quad (1)$$

i.e. for n item,

$$sim(i,j) = \sqrt{\sum_{k=1}^{n}(R_{k,i} - R_{k,j})^2} \quad (2)$$

$R_{k,i}$ is the ratings the user k gives for item i, $R_{k,j}$ is the ratings the k user gives for other item j. n is the total number of rating.

Algorithm 1: Similarity between users
Input: Preferences of user1 and user2
Output: Similarity score

1. START
2. READ si[]
3. WHILE Item!=User1 Preferences DO
4. IF ItemPref$_{User1}$==ItemPref$_{User2}$ THEN
5. Assign si 1
6. ENDIF
7. ENDWHILE
8. IF Length$_{si}$!=0 THEN
9. Calculate sim between User1 and User2 using Euclidean Distance Formula
10. RETURN 1/sim
11. ELSE
12. RETURN 0
13. ENDIF
14. END

2) Analyze Point of interest data

After collecting point of interest dataset, we have analyzed it for every user. We have discarded the irrelevant information from point of interest dataset and calculate the user's preferences like the number of hospitals, schools, universities, restaurants etc from the huge dataset of the point of interest datasets.

Algorithm 2: Point of Interest Data Analysis
Input: Point of interest dataset
Output: Ranking of the data

1. START
2. READ POI[], RANK[]
3. WHILE Item not finish for apart1 DO
4. IF Type$_{Item}$==(Univ OR Mosque OR Restu OR S. SHOP OR Park) THEN
5. Assign 1 for that type of POI[]
6. ENDIF
7. ENDWHILE
8. WHILE Item not finsh in Place1 DO
9. IF POI$_{place1}$[]>POI$_{place2}$[] THEN
10. Select Place1 and Add in RANK[]
11. RETURN RANK[]
12. ENDIF
13. ENDWHILE
14. END

3) Forecasting historical data

We have collected historical data [15] regarding the apartments. We have sorted the apartments of a specific area according to the similar historical price.

Algorithm 3: *Forecasting the historical data*
Input: list of historical data for all areas
Output: sort the apartments of an area according to the price similar to the historical data
1. **begin**
2. create an array hd
3. **for each** price of apartment1 **do**
4. **sort** distance of the price to the historical data of that area
5. store in hd
6. **end for**
7. **end**

4) Ranking Similar User

We have ranked the similar user closest to a pre-specified user using algorithm 4. We have designed the algorithm in such a way that the user matched with a specific user with the highest similarity score will stay in top.

Algorithm 5: Ranking Similar User
Input: Preference list, users
Require: Sorted list of user with a similar test with a specific user

1. START
2. SCORE[], sim
3. WHILE Other user not finish in preferences DO
4. IF Other user!= Specific user THEN
5. Calculate Similarity using Algorithm 1 and store in sim
6. IF sim>0 THEN
7. Store sim in SCORE[] for other user
8. ENDIF
9. ENDIF
10. ENDWHILE
11. READ RETURN SCORE[]
12. END

5) Recommending Apartments

To rank the apartments to recommend for a user we have generated a weighted score. We have used the concept of content-based filtering technique to generated weighted score [14].

Algorithm 4: *Top-k Real-estate recommender*
Input: Preference list, users
Output: top-k apartment

1. START
2. READ Total[], SiSum[]
3. WHILE Other user in preferences DO
4. IF Other user!= specific user THEN
5. IF Calculate similarity using algorithm 1 and store in sim THEN
6. ENDIF
7. IF sim>0 THEN
8. WHILE Other user in preferences DO
9. IF Item not in preferences THEN
10. Assign 0 in Total[] for that item
11. ENDIF
12. Multiply sim and item for other user, Add with Total[]
13. IF SiSum not in Total[] THEN
14. Assign 0 in SiSum[] for that item
15. ENDIF
16. Add SiSum[] and sim
17. ENDWHILE
18. ENDIF
19. ENDIF
20. ENDWHILE
21. READ Normalized List[]
22. READ Sort List[]
23. END

D. Complexity Analysis

For complexity analysis we have to compute the time complexity of our system which has two parts:

1. Finding top-N similar user.
2. Recommending top-k real-estate.

1) Time Complexity for Finding Top-N Similar

For calculating similarity score we have used Euclidean distance method as below:

$$r_2(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} = \sqrt{\sum_{i=1}^{2}(x_i - y_i)^2} \quad (1)$$

i.e. for n item,

$$sim(i,j) = \sqrt{\sum_{k=1}^{n}(R_{k,i} - R_{k,j})^2} \quad (2)$$

With the increasing size of users of our system, the growth of the algorithm is affected proportionally. So, the time complexity of this algorithm is linear i.e. O(n).

To rank similar users we had to sort the list. So, complexity is O(N log N).

So, total complexity is O(n) + O(N log N).

2) Time Complexity for Recommending top-k Item

For this part, we had to perform some loops, multiplication, and addition which is done linearly as the size of input affects this computation. So, time complexity is O(m).

At the last step, we had to sort the list to recommend top-k real-estate according to score. So, complexity is O(M log M).

Thus, the total complexity of our system is O(n) + O(N log N) + O(m) + O(M log M).

IV. IMPLEMENTATION

Implementation of our proposed recommender system is shown by snapshots. To get the recommendation, the users have to register and then log in to our system founded in [**17**]. The list of apartments in different cities in Dhaka is shown as Fig. 3.
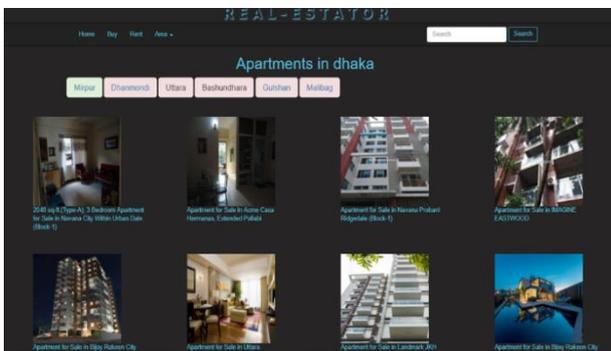

Fig.3. Apartments in Mirpur area

For the simplicity of use, we have divided places (Mirpur, Gulshan, Uttara etc.) according to the area in Dhaka. The details for a specific area appears while user clicks on it. In one word we can say that apartments in Dhaka are organized according to the area of Dhaka so that it is easy for the user to find the apartment according to their wish.

In each list, we've represented the apartments in a card view [Fig. 4]. We have only shown the pictures and name of the apartments. Further details (Name, Size of the Apartment, Price, Map of the main location of the apartment, Map of the location of nearby schools of the apartment, Map of the location of nearby hospitals of the apartment, Number of room, Number of bedrooms, Number of baths, Information of gas connection, lift, generator and many more) are given in a details page.

The system provides top-8 recommendation according to the user based on his/her selected area shown in Fig. 5. We haven't categorized our recommendation. But in our recommendation list, we give some specific information about the apartments on which users can easily understand that why and how the apartments are being ranked.

In the recommendation page, the apartments are being represented along with some extra information. There are being shown the number of university, school, restaurant, and banks around 500 meters for each apartment which we got from the POI database of google map.
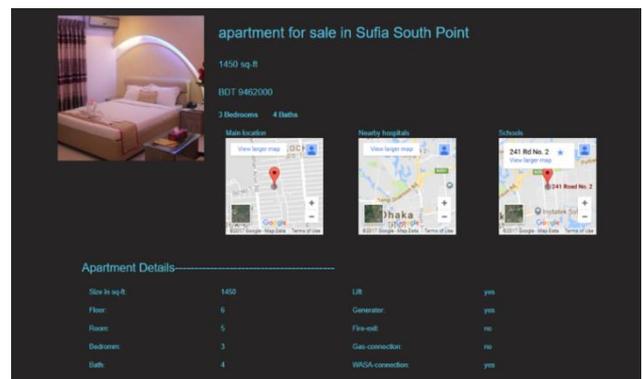
Fig.4. Details of a selected apartment (Sufia South Point)
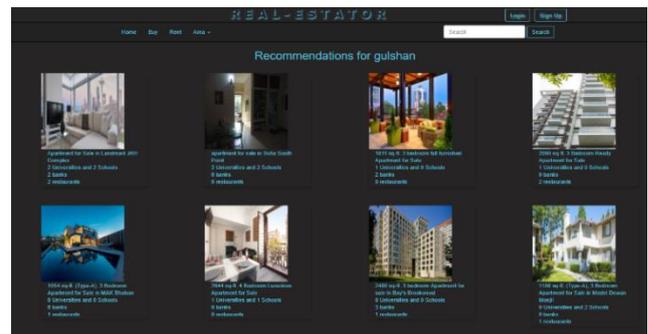
Fig.5. Top-8 recommended apartments for a selected area (Gulshan)

V. THE EXPERIMENTAL RESULT AND ANALYSIS

Users can get the recommendation by searching for a specific location. After searching a specific location our system finds out the apartments in that area at minimum cost considering his-historical data and surrounding environments. For recommendation, the system analyzes all apartments but recommends top-k apartments. We use ratings from the POI database for user preferences. Even we took query about price range and preferable facilities from users for a better recommendation. The system can also recommend based on those user queries. The user can also rate the apartments.

Performance of our proposed recommendation system has evaluated using precision. Precision is measured as a fraction of actually relevant recommended items to the user. According to the rating of the user, the real estate items are recommended by the system. But among all, the correctly recommended items are suggested with the price range and surrounding facilities queries by the users.

According to the 1st user Uchchash:

The number of total recommended items=12
The number of correctly recommended items=8
So, the system precision is=8/12=.66

Precision values of five users are enlisted in Table 1 and represented in Fig. 6.

Table 1. Calculating Precision for different users

| User	Total recommended items	Correctly recommended items	Precision
Uchchash	12	8	.66
Samiran	12	9	.75
Shemonti	12	7	.783
Antor	12	9	.75
Tandra	12	10	.833
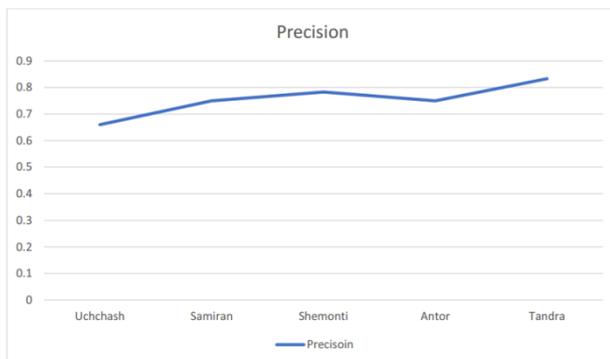
Fig.6. Graphical representation of the Precision value

VI. CONCLUSION

In this paper, we have developed a real-estate recommendation system including a user-friendly graphical interface for easy navigation, consideration the point of interest for every real-estate apartment, the recommendation of top-k real-estate with ranking and predicted score for a particular user. Considering one hundred real-estate and fifty users we have tested our proposed system using collaborative filtering through extensive experiments on synthetic and real data. Our proposed system can generate satisfactory recommendations to the users with more accuracy and scalability. At the end, the experimental result shows that the proposed method improves the accuracy significantly.

Limited Datasets are the main limitations of our proposed systems. In this systems big data is not used which is one of the mandatory factors for a recommendation system. Big data analysis can give a perfect result to the users. The more a system recommends correctly the more the users get benefitted. For example, amazon, youtube etc. use user preferences, their ratings and many other factors to recommend things through the big data analysis. We've used google map API but we could not use live google map location for the users which is the major disappointing issues for this system.

For further research, we will collect a large dataset of user preferences, also many more areas and use big data analysis for better efficiency. As data are fetched dynamically through API, it needs to enhance the speed and correctness of the application for providing a better recommendation.

REFERENCES

[1] Barua, Suborna, et al. "Housing Real Estate Sector in Bangladesh Present Status and Policies Implications." *ASA University Review*, vol. 4, no. 1, Jan. 2010, www.asaub.edu.bd/data/asaubreview/v4n1sl18.pdf.

[2] Atisha Sachan and Vineet Richariya," Survey on Recommender System based on Collaborative Technique", Department of Computer Science And Engineering, *International journal of innovations in engineering and technology (IJIET)*, ISSN: 2319-1058, vol.2, issue 2, pp1-7, April 2013.

[3] Prem Melville and Vikas Sindhwani," Recommender System", IBM T.J. Watson Research Center, Yorktown Heights, pp 1-18.

[4] J. Gupta and J. Gadge, "Performance analysis of recommendation system based on collaborative filtering and demographics," *2015 International Conference on Communication, Information & Computing Technology (ICCICT)*, Mumbai, 2015, pp. 1-6. doi: 10.1109/ICCICT.2015.7045675

[5] F. Gao, Y. Li, L. Han and J. Ma, "InfoSlim: An Ontology-Content Based Personalized Mobile News Recommendation System," *2009 5th International Conference on Wireless Communications, Networking and Mobile Computing*, Beijing, 2009, pp. 1-4. doi: 10.1109/WICOM.2009.5300815

[6] P. Mathew, B. Kuriakose and V. Hegde, "Book Recommendation System through content based and collaborative filtering method," *2016 International Conference on Data Mining and Advanced Computing (SAPIENCE)*, Ernakulam, 2016, pp. 47-52. doi: 10.1109/SAPIENCE.2016.7684166

[7] Anand Shanker Tewari, Abhay Kumar, Asim Gopal Barman," Book Recommendation System Based On Combine Features Of Content Based Filtering And Association Rule Mining",*IEEE International Advance Computing Conference(IACC)* ISSN:47992572, pp502 14th augest2014.

[8] Chhavi rana, sanjay kumar Jain," Building a Book Recommender system using time based content Filtering", University Institute of Engineering and Technology, ISSN: 2224-2872, Issue 2, Volume 11, 6 February2012.

[9] Z. Chang, Md. S. Arefin, Y. Morimoto, "Hotel Recommendation Based On Surrounding Environments" , *2nd IIAI International Conference on Advanced Applied Informatics, 2013*

[10] Ding, Zhijun, et al. "Objectives and State-of-the-Art of Location-Based Social Network Recommender Systems." *ACM Computing Surveys*, vol. 51, no. 1, 2018, pp. 1-28.

[11] V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang, "Collaborative location and activity recommendations with GPS history data", *In Proc. of the 19th international conference on Worldwide web*, pp. 1029-1038, 2010.

[12] K. Kodama, Y. Iijima, X. Guo, and Y. Ishikawa, "Skyline queries based on user locations and preferences for making location-based recommendations", *In Proc. of LBSN*, 2009, pp.9-16.

[13] Wang, Fan, et al. "Mining user preferences of new locations on location-based social networks: a multidimensional cloud model approach." *Wireless Networks*, vol. 24, no. 1, 2016, pp. 113-125.

[14] Chowdhury, Chondrima, et al. "Developing a framework for recommending TV shows." 2017 6th International Conference on Informatics, Electronics and Vision & 2017 7th International Symposium in Computational Medical and Health Technology (ICIEV-ISCMHT), 2017.

[15] EzEstate (2017, Jan 10). Historical data, Available at: https://esestate.com/real-estate-historical-data.html?standard

[16] Lee, Yunkyoung, "RECOMMENDATION SYSTEM USING COLLABORATIVE FILTERING" (2015). Master's Projects. 439. http://scholarworks.sjsu.edu/etd_projects/43

[17] Gitlab, "Real Estate Recommendation System", https://gitlab.com/uchchash/realestate_recommendation, hosted on November 2017.

Md. Sabir Hossain received his bachelor degree in computer science and engineering from Chittagong University of Engineering & Technology (2015) with an outstanding result. He has also completed his M.Sc. degree in computer science and engineering in 2019 from the same university. His research interests are algorithmic complexity analysis, data mining, machine learning, big data, and software engineering. He is currently working as a faculty member (Assistant Professor) in the same department of his university. He has a number of publications in international conferences as well as journals. Besides teaching and research activities, he also works as an independent freelancer.

Mohammad Shamsul Arefin received his B.Sc. Engineering in Computer Science and Engineering from Khulna University, Khulna, Bangladesh in 2002, and completed his M.Sc. Engineering in Computer Science and Engineering in 2008 from Bangladesh University of Engineering and Technology (BUET), Bangladesh. He has completed Ph.D. from Hiroshima University, Japan. He is currently working a Professor and Head in the Department of Computer Science and Engineering, Chittagong University of Engineering and Technology, Chittagong, Bangladesh. His research interest includes Data privacy and Data Mining, Cloud Computing, Big Data, IT in Agriculture, and OO system development.

Authors' Profiles

Uchchash Barua has completed his B.Sc. degree from the Department of Computer Science and Engineering from Chittagong University of Engineering & Technology (CUET) in 2017. Now, He is looking forward to an opportunity for working in a challenging environment where he can utilize his knowledge for developing his career and for the growth of the organization.