Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 171 (2020) 197–206

Procedia Computer Science

www.elsevier.com/locate/procedia

Third International Conference on Computing and Network Communications (CoCoNet'19)

Smart Rent Portal using Recommendation System Visualized by Augmented Reality

Shashank Mouli Satapathy[a,*], Rutanshu Jhaveri[a], Ujjwal Khanna[a], Ashish Kumar Dwivedi[b]

[a]*School of Computer Science and Engineering*
Vellore Institute of Technology, Vellore
Tamil Nadu, India.
[b]*Department of Computer Science and Engineering*
Gayatri Vidya Parishad College of Engineering
Madhurawada Visakhapatnam, 530048, India

Abstract

Context: Locating products or services on-line that meet users want is more and more troublesome thanks to the massive pool of decisions to think about before inbound at the required one. A user might pay a substantial quantity of your time exploring varied on-line resources to find things that work his needs. Moreover, users might not forever categorical their preferences in an exceedingly manner that simply matches them to things that might meet them. *Objective:* Searching out things on-line has been done principally through information queries that come back an inventory of the foremost appropriate things. Recommendation systems technology may be applied to ease the task of locating the desired things on-line. *Method:* This study proposes a recommender system allows that permits that allow users to hold out a preference-based cooperative filtering search on rental properties and enables them to refine those preferences based on shallow learning. The room also is visualized using AR and Vuforia. *Results:* This recommendation approach has been shown to supply additional correct search results. The system was developed as an Internet application victimization Handlebars for front-end and Nodes- ExpressJs for Back-end. *Conclusion:* The system performs better than existing algorithms and predicts better in memory-based approach.

Keywords: Augmented Reality; Centralized; Collaborative Filtering; Properties; Recommender System.

1. Introduction

A plethora of websites exists that are used daily for searching properties in and around a city[? ?]. A Website or an Android application is convenient to use and user friendly. It is able to target audience and reach to people who just need an access to a mobile phone or a minimal Internet connection. Many students and other people in educational

* Corresponding author. Tel.: +91-8290979509.
E-mail address: shashankamouli@gmail.com

10.1016/j.procs.2020.04.021

institutions have difficulty of finding rooms around them that are available for rent. Rental apartments in and around these institutions do not have any form of medium where they can display their houses for rent. This causes them to rely on word of mouth and huge hoardings, which are viewed by people around the university.

Recommendation systems are software tools and methods, which can offer suggestions for items to be of usage to a user. Users rely on these suggestions to make various decisions include what movies to watch, what news articles to read, what apartment to rent, etc. These systems have proved to be useful in helping users deal with overwhelming amount of information while they search for various items online[???]. This study will help in the development of a rent portal based on a recommender system for rental properties.

Many websites are available such as NestAway, CoLive, No Broker, Quickr PG and others; however, the main flaw of these websites is their lack of robustness and an interactive User Interface. These websites do not even target small property owners and apartment dealers. Nor, do they offer any smart solutions to the existing problems. Hence, a need for a website falls into picture. Tenants who are a part of these apartments also can post an ad with permission of the property owner if they require one more roommate.

The current preferred collaborative filtering models build use of one thing referred to as embedding matrices. Embedding matrices contain third-dimensional data. For example, allow us to say we've got an embedding matrix for rent homes with 3 factors. it should be the case that these 3 factors correspond to parking lot provided, 24X7 Electricity and Wi-fi. Of course, these factors may correspond to something (and aren't essentially straightforward to interpret) and embedding matrices usually contain several factors. These matrices ar updated once coaching a collaborative filtering model. using the dataset collected, with our standard collaborative filtering technique we might have an embedding matrix for the users and homes.

The research work carried out in this paper proposes a recommender system allows that permits that allows users to hold out a preference-based cooperative filtering search on rental properties and enables them to refine those preferences. Cosine similarity is used for the collaborative filtering techniques that are used in this paper. This allows similar products to be grouped together where we can easily recommend them and show it to the user. This recommendation approach has been shown to supply additional correct search results. Additionally, the system was tested to determine that it performed as designed. It is also practically difficult for someone to be able to get a visual delight when looking at rooms. Often these rooms have spatial dimensions, which cannot be captured using photos only. Hence, what we propose is to build a 3D Model around which the user can see the property in Augmented Reality Space.

2. Related Work

Jinhao et al.[1] have proposed a method for real state recommendation based on the users preference on the location and neighboring facilities. The main characteristic of the paper is that they consider the inspiration of coexistence of some services in the adjoining areas. The issue with method is that query time rises with the increase of distance between query point and requested target facility. The reason for this that with the rise in distance, we need to consider more items in calculation. In this paper, we have used cosine similarity with collaborative filtering on location and neighboring amenities. This method reduces the time for fetching the appropriate results.

Nicolas Nahimana Guy[2] have implemented a recommender system for rental properties. They use user-initiated critiquing systems focusing on the user preference elicitation, system recommendation, user feedback and selection of desired items. The issue with this that If the number of attributes in which users express preferences increasing, the user efforts also increase making it harder to the system to be used. The main objective of this recommender system was to facilitate locating rental properties. Therefore, if the number of attributes of rental properties increases, this may undermine the primary objective of building such a system. Consequently, the researcher limited the number of attributes on which users can express preferences to only five.

Ng et al.[3] have implemented a vacation rental portal. To improve the penetrating efficiency, three techniques were recommended in this work: images-based cosine similarity calculation, textual explanation established jaccard similarity calculation, and the combination of both preceding mentioned techniques. It was presented that recommendation based exclusively on image comparisons or textual description-based similarities are possible. However, it seems that the traveling choices of users could not be seized totally using either one. The images-textual hybrid recommendation system accomplishes an enhanced result, as this technique is able to catch the data miss out by either

image or textual data. Nonetheless, this method is only appropriate for users who have pasts. Therefore, our rent portal does not require any user history for the recommendation system to work.

The basic algorithms used in the recommender systems are assessed by Alrawhani et al.[4]. The accuracy of the results are not on point. Hence, our portal implements matrix factorization techniques and collaborative filtering to the right and appropriate results.

Shweta Athalye[5] comprises the characteristics of both Content-Based and Collaborative centered filtering methods, considering the advantages of the hybrid method and tries to rectify the limitations of both the methods to deliver proactive references to a newsreader. Therefore, they use an Android Client, which is not efficient for web applications.Our project uses Web Client that is suitable for all cross-platforms operations.

Xiaofang et al.[6] have implemented a user-oriented real estate recommendation system. Although, the study has some limitations. Firstly, the house buyers' choices are affected by many factors. They also depend on the distance to a location and other environmental aspects such as safety, amenities, and convenience amenities, etc. Therefore, they only accepted the resemblance measurement of site accessibility, and only data were accessible for the current research. Secondly, they have used a different database for the recommendation system for sellers and another for tenants. Our project works on all distance and environment factors. Also, uses a single database for recommendation of sellers and tenants.

Zhang et al.[7] have recommended various answers to make a quality recommendation. The techniques mentioned in this paper are related to various collaborative filtering procedures, which include the matrix-factorization techniques, and the neighbors based techniques. Therefore, they have not applied these techniques on big data sets. We have applied our collaborative filtering techniques on big data sets of sellers, tenants and properties.

Alhijawi et al.[8] has mentioned about improving collaborative filtering recommender systems using semantic information. Semantic information in this paper is based on what people have previously rated and what information is provided by the items so that the recommendation system can be built on top of that, however this is prone to errors such as users adding in wrong data or rating or the person who uploads giving false information. Our systems avoids this by doing a background research on every property owner and binding them with few legalities. The proposed algorithm also views the low rated items and gives them lesser importance in the recommendation system.

Nawrocka et al.[9] has mentioned how machine learning is applied to recommendation systems and what results are obtained using this form of learning. They have taken various type of machine learning algorithms and distinguished it. However, in this case cannot be also mentioned for all forms of systems. In the analysis done, the data sets were ensured a certain amount of minimum ratings issued by the users. For our system, we have included a hybrid method of Collaborative filtering along with user ratings hence allowing us to overcome this flaw.

Ajay Singh et al.[10] has mentioned about creating an effective recommender system using machine learning based framework precisely focusing on mahout as a recommender platform. Our method uses Collaborative Filtering, which has more features in the ML tool kit compared to the paper's method. The algorithm along with K - Means Clustering would allow our system with large data sets, which is not covered in this study.

3. Recommendation Systems

3.1. Progression of recommender system

Automated recommender systems supported cooperative filtering emerged within the Nineties. a number of these enclosed Ringo for music, BellCore Videos Recommender for movies, and merry andrew for jokes, among others. Maybe the foremost identifiable business application of recommender system is Amazon. supported the user purchase history, browsing history, the present item the user is viewing, and different users behaviour, Amazon will advocate things for the user to consider buying[11].

Recommender technology has gone on the far side cooperative filtering to incorporate content-based, Bayesian logical thinking and case-based reasoning strategies. analysis on recommender systems gained momentum with the launch of the Netflix Prize, a one-million-dollar reward for analysis that might improve by 100% the accuracy of Netflix recommendations for movies.

3.2. Recommendation System Functions

Recommender systems play a range of roles. These functions fall under 2 categories: the roles recommender systems play on behalf of the service supplier and for the end-user)[12], a service supplier might need to use a recommender system to attain the following:

1. **Augmented sales:** The service supplier would really like to sell a lot of things than those he may sell with none recommender system. This goal is achieved as a result of the suggested things suit clients wants. the first purpose of employing a recommender system then is to extend the conversion rate.
2. **Diversity of sold items:** R Recommendation systems conjointly facilitate users notice things they'll not have discovered within the absence of a certain recommendation for those things. This way, a service supplier will sell things that square measure unpopular normally, however that will suit specific users.
3. **Increase user fidelity:** Users square measure a lot of probably to return a web site that recognizes returning users and treats them as special guests. Since recommender systems use info from previous user behavior (ratings), the more a user interacts with the system, the more a user relates with the system, the more polished his user preference model becomes. While there are various ways of categorizing the inputs from the targeted client, one compelling set of groups evolves from the customers method toward providing the input[13].
4. **A better understanding of users' desires:** Recommendation systems develop an outline of users preferences collected either expressly or implicitly. The service supplier will utilize this data to attain different goals like rising the management of the items production or stock.
5. **Find all useful items:** this involves recommending all the objects that will meet users desires. It's principally common once the amount of articles to counsel is little and in mission-critical things like medical and monetary applications.

4. Methods of Recommendation Systems

Recommender systems have many recommendation methods, the most well-known and used nowadays in the recommender systems are these three following general methods: content based filtering, collaborative filtering, and case based filtering. Each method has a unique effectiveness and accuracy regarding recommendation based on the applying parts and the action stages[14]. So, it is important to find the best method to implement in our system based on the characteristic of each method, we need to find one or two of these methods to be useful for our Real Estate Recommender system.

4.1. Collaborative Filtering Approach

Collaborative filtering is extremely well-known technique that has been enforced by some recommender systems. Cooperative filtering is the filtering where patterns exploitation techniques involving collaboration among multiple agents, viewpoints, knowledge sources, etc. take place. Applications of cooperative filtering generally involve very Brobdingnagian information sets. This approach has been enforced to E-commerce and Internet applications wherever the attentiveness on user knowledge, etc.

Collaborative filtering is manufacturing suggestions (filtering) supported the user interests by gathering alternative users preferences or rating info for things (collaborating). the elemental theory of the cooperative filtering approach is that if a user one has the similar style as a user a pair of on associate item, user one is presumably has user 2's opinion on a unique item x than to has the style on x of an individual chosen haphazardly.

Collaborative filtering approach works by building the information of preferences for things by users. a replacement user is matched against the information to search out neighbors (other users) WHO have previously had similar interests because the new user. things that the neighbors like square measure then suggested to the new user, as he can possibly additionally like them.

Cosine similarity in collaborative filtering:

$$simil(x,y) = cos(\vec{x},\vec{y}) = \frac{\vec{x}.\vec{y}}{||\vec{x}|| \times ||\vec{y}||} \tag{1}$$

In the system we propose, the product is the property, which has been uploaded by the agent, the user is able to rate it and along with that the user uploads his own preferences on the basis what he prefers[15]. The recommendation then uses collaborative filtering after a K-Means Clustering (if the data set is too large). The filtering allows a spatial matrix to be created, where user choice and property amenities are drawn out. Later such recommendations are given to the user.

4.2. Content-Based Filtering

The Content-based filtering approaches make recommendations by analysing the description of the items rated by the users and the description of the items to be recommended. In content-based filtering, keywords are used to define the items and a user profile is formed to designate things the user may like. The similarity of products is calculated based on the features associated with the compared items. Here a new aspect is added as temporal dimension with a kiosk value for each element, which gets update with time and thereby improves the recommendation process[16]. If Mary likes watching comedies or movies featuring Denzel Washington, it may be reasonable to recommend to her other comedies or movies starring Denzel Washington.

4.3. Hybrid Recommendation System

A mix of each techniques may be utilized in various ways like by creating content-based and conjointly collaborative -based predictions clearly and mixing the particular algorithms in barely one model. Various studies are distributed so as to check the particular potency within the hybrid technique with all the important cooperative and conjointly content-based strategies and conjointly hybrid strategies are shown to produce much better accuracy. This might even be utilized in order pass though the restrictions of ancient Recommendation system like cold begin and also the scantiness drawback. Netflix is really AN illustration of hybrid system. The disadvantage of hybrid recommendation system may be resolved by a replacement theorem network design.

5. Proposed System

We have deployed this system on localhost of our computer system. In PC networking, localhost is a hostname that implies this PC. It is utilized to get to the system benefits that are running on the host by means of the loopback arrange interface. Utilizing the loopback interface sidesteps any nearby system interface equipment.

The IDE for the project is WebStorm 2018. WebStorm is a cross-platform IDE primarily for web, JavaScript and TypeScript development. Many of JetBrains other IDEs include the feature set of WebStorm via plugins. The IDE for the project is WebStorm 2018. WebStorm is a cross-platform IDE primarily for web, JavaScript and TypeScript development. Many of JetBrains other IDEs include the feature set of WebStorm via plugins.

You can now utilize Node.js on Windows Subsystem for Linux to run and investigate your Node.js application and for the various improvement assignments in WebStorm.

- WebStorm 2018.2.1
- Build #WS-182.3911.37, built on August 7, 2018
- Subscription is active until May 6, 2019 For educational use only.
- JRE: 1.8.0_152-release-1248-b8 amd64 JVM: OpenJDK 64-Bit
- Server VM by JetBrains s.r.o Windows 10 10.0

Nodemon runs the app without the need to constantly exit and run it repeatedly via the terminal. nodemon is a tool that helps develop node.js based applications by automatically restarting the node application when file changes in the directory are detected. Nodemon does not require any additional changes to your code or method of development. Nodemon is a replacement wrapper for node, to use nodemon replace the word node on the command line when executing your script.

As shown in Fig 1, we run our web application on Node Js and Express JS backend server. This server is deployed on Heroku and saves all the data on a MongoDB instance. The frontend which displays all the dynamic data from the

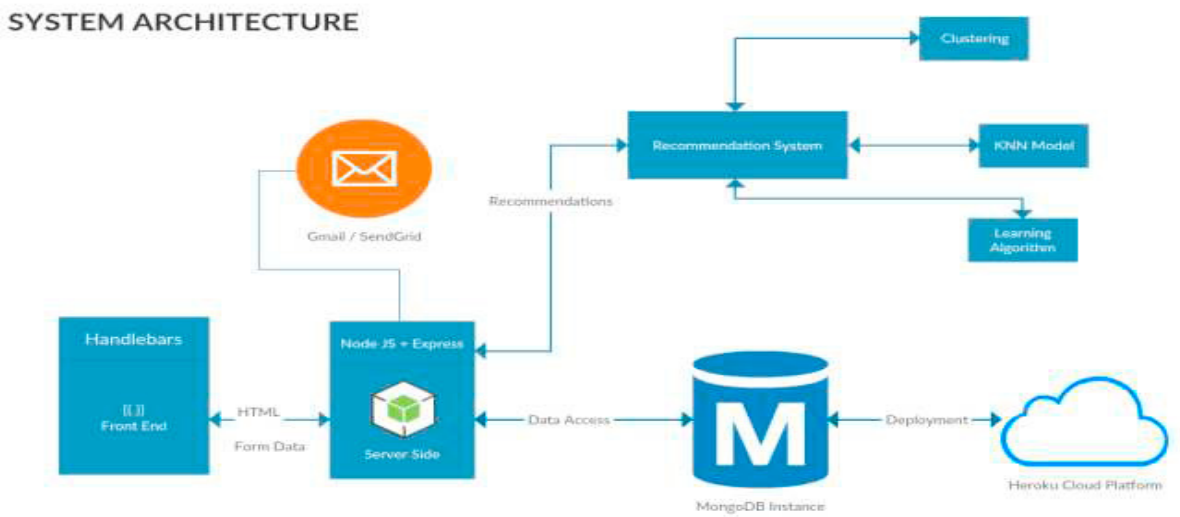

Fig. 1: System Architecture of Proposed System

sever is done using Handlebars which is very lightweight and makes the code clean and easy to use. The recommendation system works in coherence with the Backend and makes changes based on the data access to the MongoDB instance. The KNN model using clustering techniques, which allow working on large data sets[17]. This is very similar to the principle of case-based reasoning (CBR) and the personal recommendation cases[18].

As discussed earlier, the current preferred collaborative filtering models build use of one thing referred to as embedding matrices. Embedding matrices contain third-dimensional data. For example, allow us to say we've got an embedding matrix for rent homes with 3 factors. it should be the case that these 3 factors correspond to parking lot provided, 24×7 Electricity and Wi-fi. Of course, these factors may correspond to something (and aren't essentially straightforward to interpret) and embedding matrices usually contain several factors. These matrices ar updated once coaching a collaborative filtering model. using the dataset collected, with our standard collaborative filtering technique we might have an embedding matrix for the users and homes. Decision making is done based on various parameters such as ratings, reviews, amenities present in the property, pricing of the property and other factors.

The size of the matrix would be the amount of users or homes by the amount of things we decide. with regard to picking the amount of things within the embedding matrices, that needs some trial-and-error. the amount of embedding factors was set to fifty. Before the beginning of coaching, the values among these embedding matrices ar indiscriminately initialized. During training, these values are updated in service of reducing the loss (i.e., making the predicted ratings more similar to the actual ratings).

During each iteration of training, for each user's rating of a house, the dot product of the corresponding vectors are taken. This dot product is the expected rating. The dot product ist aken for each single user's rating of each rental house rated (Note: homes that weren't rated by a moving-picture show square measure set to 0), and therefore the expected ratings square measure compared to the particular ratings. Then, random gradient are used to update the values at intervals the embedding matrices in commission of reducing the loss operate. additionally to the embedding matrices, progressive collaborative filtering models contain a bias term, that is actually to account certainly users that perpetually provide either a lot of high or low ratings or rooms that overall square measure given a lot of high or low ratings.

6. Interface Design

6.1. User Interfaces

The user interface provides an easy access to the facilities offered by the web app. The color scheme used will bring attention to detail and will bring the important aspects over to the forefront. Bootstrap was used to create interactive and responsive buttons with dynamic layouts that change in real time[19].

6.2. Software Interfaces

This web app is developed in HTML, CSS and JavaScript. Bootstrap framework is used for the front-end layouts and dynamic constraints that will effortlessly fit for every screen. MongoDB is used for the back-end as it delivers robust and intuitive procedures to store and manipulate date, CRUD operations and create model schemas, which uses NoSQL.

6.3. Communication Interfaces

Bcrypt is a password-hashing task designed by Niels Provos and David Mazires, centered on the Blowfish cipher, and introduced at USENIX in 1999. Besides integrating a salt to protect against rainbow table strikes, Bcrypt is a versatile capacity: after some time, the redundancy check can be expanded to make it slower, so it stays strong to savage power seek assaults even with rising calculation control. International ID is au-thentication center product for Node.js. Massively adaptable and particular, Passport can be subtly dropped in to any Express-based web application. A total arrangement of systems bolster check utilizing a username and secret key, Facebook, Twitter and so forth.

7. Implementation Design

The interface of Rent Portal with recommendation system with cooperative filtering approach is the most vital part. Nowadays, individuals are concerned with package interface design before trying to find the functionalities of the system. Thus, to form our system more enticing and dynamic in look, we've used graphical UI as the interface of selection for planning our system. Below, we will be show the foremost vital functions of the system.

7.1. Home Page of the portal

The Home page is the landing page of the web application and would doing a basic filter for the website such as the BHK number or price range. This is shown in Fig 2 below and can be viewed by any user such as an admin, patron or property owner.

7.2. Property Search

User will search his/her most popular property, wherever he/she will key within the preference of the property like (City, State and ZIP). The user can also key within the Tags (keywords) relating to any property to ease the search operate, these Tags are the cases for the case-based approach, these cases are saved within the information for future work by different user trying to find the search preference. This is shown below in Fig 3.

7.3. Search Results

The computer program is our main mission for this project, once the user click the search button the result showed are from the case-based reasoning and traditional search operation[20]. Cooperative filtering offers best result for the user and suggests these results to the user, if there's no result for the cooperative filtering, the system can show the result for the traditional search function, wherever all obtainable properties are list it for the user as shown below in Fig 4.

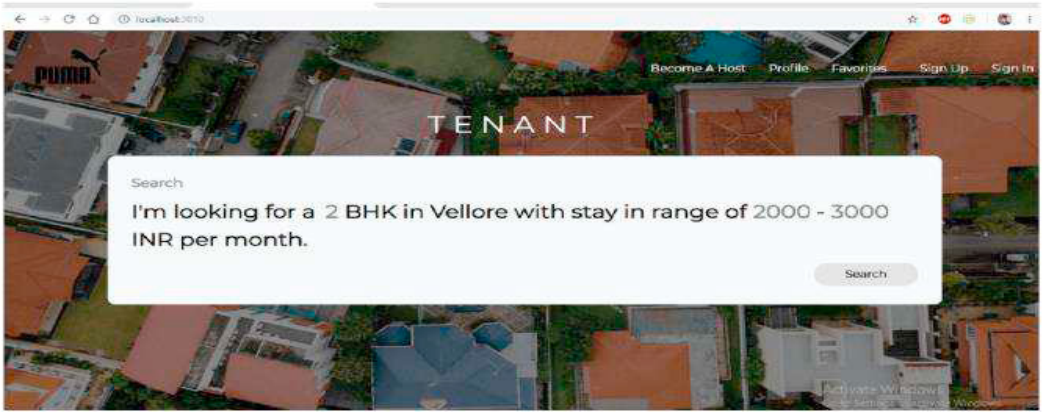

Fig. 2: Portal Home Page

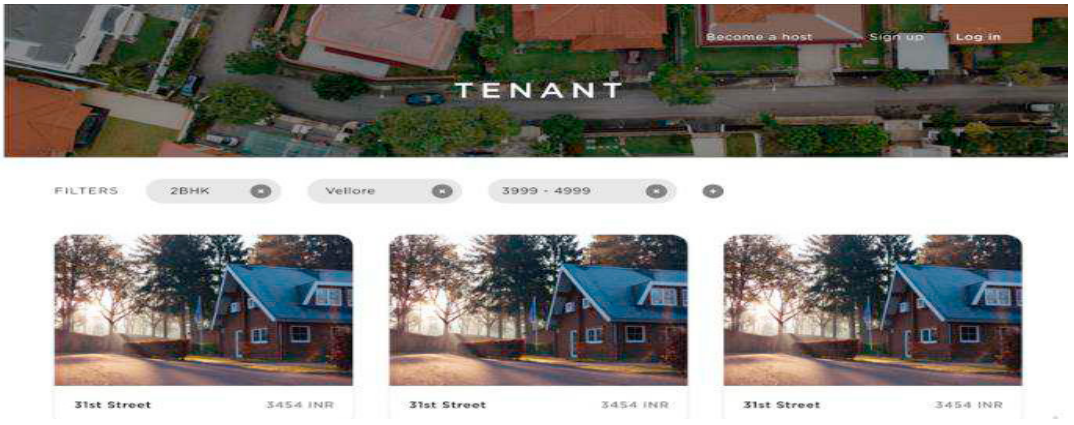

Fig. 3: Property Search Page

Fig. 4: Property Search Result

7.4. Visualization using Augmented Reality

Augmented truly gives us stunning experiences. Whether it is playing games on our phone, using filters on social media, or running simulations, this new tool has enabled us to interact with the environment around us and understand it even better.

To get a better understanding of spatial sense, AR can be a very powerful tool. For our project, we have used AR along with Vuforia to set image targets of properties in Unity. At first, the property room image is clicked. Later this is set as the image target in Vuforia for that particular room. A 3D model is prepared for that room using 3DS Max. We then place this model on top of the image target. This is later imported in the mobile application and the image target is displayed on the website.

Whenever the user using our mobile app, places the camera on the target image on the web app, they are able to view the 3D Model of the room. This allows getting a feel how it looks and getting the spatial understanding of how it is when the user enters the actual room.

8. Result Analysis

Table 1 displays the comparison of proposed system with existing work. Below we see all the works and how we have provided a better solution. Our system mainly consists of a recommendation system, which provides a novel solution over the current works. The system performs better in larger data sets compared to the existing algorithms that are present and given in the study.

Table 1: Comparative Analysis of Existing Work

Product Name	Issues Occurred	Solution Provided by our Portal
Airbnb	No option to substitute to a vacating tenant No recommendation system	Tenant Substitution is provided
Nest Away	No learning algorithm implemented,	Recommendation system based on collaborative filtering
Quickr PG	Dull and Unattractive UI. No machine learning implementation	Intuitive and user-friendly UI Design, with a recommender system.
Colive	A Unappealing UI, No recommendation system. Less market penetration	Attractive and Material design with machine learning

Table 2 displays an analysis done with existing algorithms and how we have improved on it.

Table 2: Comparative Analysis of Existing Work

Existing Algorithms	Drawbacks	Our Algorithm's improvement
Euclidean Distance	Not flexible on various other systems	Cosine Similarity offers more flexibility and is not required to be symmetric or fulfill triangle inequality
Non negative Matrix Factorization	Layers cannot be added easily to this algorithm when you wish to further your neural net in deep learning	We use Probabilistic Matrix Factorization implemented with shallow learning techniques, which allowed us for the system to have more neural nets.

9. Threats to Validity

Our system is meant for a situation currently that is present around an educational institution. The following are the cases where there may be certain threats to validity:

- The institution curbs students from staying outside and causes a loss of market for the system. This will cause fewer people opting to stay outside and may lead failure of the application.
- Monopolizing giant companies that may venture into this market space and cause disruption leading to the application no longer remaining relevant. The novelty of the web application would come in play and lead to various changes in its functioning.

- Learning Algorithm and techniques such as Collaborative Filtering may not be very functional in situations where users rate bizarrely and have a certain bias towards a particular form of property owner.
- Bot attacks that may manipulate the dataset for the recommendation system and cause it to predict bogus properties.

10. Conclusion and Future Work

This project would be able to influence the lives of students where they can view and access relevant data at the tip of their fingers. Students from other universities could also adopt the same concept and change the way in which this system works for different environments.

We aim to drive the application through technology. This will allow users to get hands on experience of technologies such as VR and Block-chain techniques. VR can be used to visualize properties and its layout for any user. Having the power to move objects around in your own house without actually having any form of physical component attached to it also is a possibility. Block-chain comes into consideration when we are looking at payment methods done by users on the site and also a decentralized tenant and landlord management system.

References

1. Jinhao, X., Arefin, M.S., Zhiming, C., Morimoto, Y.. Real estate recommender: Location query for selecting spatial objects. In: *2012 Third International Conference on Networking and Computing*. IEEE; 2012, p. 278–282.
2. Guy, N.N.. *A Recommender system for rental properties*. Ph.D. thesis; Strathmore University; 2017.
3. Ng, P.C., She, J., Cheung, M., Cebulla, A.. An images-textual hybrid recommender system for vacation rental. In: *2016 IEEE Second International Conference on Multimedia Big Data (BigMM)*. IEEE; 2016, p. 60–63.
4. Alrawhani, E.M., Basirona, H., Sa'ayaa, Z.. Real estate recommender system using case-based reasoning approach. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* 2016;**8**(2):177–182.
5. Athalye, S.. Recommendation system for news reader. 2013.
6. Yuan, X., Lee, J.H., Kim, S.J., Kim, Y.H.. Toward a user-oriented recommendation system for real estate websites. *Information Systems* 2013;**38**(2):231–243.
7. Zhang, R., Liu, Q.d., Wei, J.X., et al. Collaborative filtering for recommender systems. In: *2014 Second International Conference on Advanced Cloud and Big Data*. IEEE; 2014, p. 301–308.
8. Alhijawi, B., Obeid, N., Awajan, A., Tedmori, S.. Improving collaborative filtering recommender systems using semantic information. In: *2018 9th International Conference on Information and Communication Systems (ICICS)*. IEEE; 2018, p. 127–132.
9. Nawrocka, A., Kot, A., Nawrocki, M.. Application of machine learning in recommendation systems. In: *2018 19th International Carpathian Control Conference (ICCC)*. IEEE; 2018, p. 328–331.
10. Singh, A.V., Sharma, M., et al. Building an effective recommender system using machine learning based framework. In: *2017 International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions)(ICTUS)*. IEEE; 2017, p. 215–219.
11. Fu, M., Qu, H., Yi, Z., Lu, L., Liu, Y.. A novel deep learning-based collaborative filtering model for recommendation system. *IEEE transactions on cybernetics* 2018;(99):1–13.
12. Gao, T., Li, X., Chai, Y., Tang, Y.. Deep learning with consumer preferences for recommender system. In: *2016 IEEE International Conference on Information and Automation (ICIA)*. IEEE; 2016, p. 1556–1561.
13. Schafer, J.B., Konstan, J.A., Riedl, J.. E-commerce recommendation applications. *Data mining and knowledge discovery* 2001;**5**(1-2):115–153.
14. Zhang, X., Wang, H.. Study on recommender systems for business-to-business electronic commerce. *Communications of the IIMA* 2005; **5**(4):8.
15. Jiang, W., Zhang, M., Zhou, B., Jiang, Y., Zhang, Y.. Responsive web design mode and application. In: *2014 IEEE Workshop on Advanced Research and Technology in Industry Applications (WARTIA)*. IEEE; 2014, p. 1303–1306.
16. Rana, C., Jain, S.K.. Building a book recommender system using time based content filtering. *WSEAS Transactions on Computers* 2012; **11**(2):2224–2872.
17. Thomas, A., Sujatha, A.. Comparative study of recommender systems. In: *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*. IEEE; 2016, p. 1–6.
18. Sun, J., Li, L., Zhang, R.. Personal recommender system based on case-based reasoning. In: *2009 International Conference on Industrial and Information Systems*. IEEE; 2009, p. 178–181.
19. Majid, E.S.A., Kamaruddin, N., Mansor, Z.. Adaptation of usability principles in responsive web design technique for e-commerce development. In: *2015 International Conference on Electrical Engineering and Informatics (ICEEI)*. IEEE; 2015, p. 726–729.
20. Saaya, Z., Smyth, B., Coyle, M., Briggs, P.. Recommending case bases: Applications in social web search. In: *International Conference on Case-Based Reasoning*. Springer; 2011, p. 274–288.