Online Study and Recommendation System

Junnutula Meghanath Reddy
Texas A&M University
College Station, Texas
meghz17@neo.tamu.edu
9797395339

Tengyan Wang
Texas A&M University
College Station, Texas
michealwty@neo.tamu.edu
9795879818

ABSTRACT

Online Study and Recommendation system is a public or private destination on the internet that addresses the individual needs of its members by facilitating peer-to-peer study environment. In this paper we describe the basic idea of such a system to be developed as a part of the Computer Supported Cooperative Work graduate course. It is a single, robust, secure and integrated system that provides course recommendation and features for collaborative study. We discuss the basic approach, implementation, recommendation algorithms and finally the results of the user study. Most of today's study systems are not safe and neither do they provide good privacy for the students to study in a distributed environment. Thus, we have created such a system to employ individual and group learning process with several features supporting the activity.

General Terms

CSCW

Keywords

Recommendation, distributed, education, online.

1. INTRODUCTION

In today's world, there are many systems which provide an environment for students to study online but rarely are they provided with good privacy and integrity for information exchange. Most of them have been forums or blogs where specific groups have not been given prime importance. The Online Recommendation and Study system mentioned in this paper introduces a unique distributed scenario for collaboration between students all over the world to study, share and make discussions. It is a public or private destination on the internet that addresses the individual needs of its members by facilitating peer-to-peer study environment. The system being a web-based application would provide great consistency and conformity among the group using the distributed system. Our system emulates the traditional E-learning system study environment with more preference given to the individuals and their preference.

E-learning is an inclusive term that describes educational technology that electronically or technologically supports learning and teaching. E-learning may either be synchronous or asynchronous. Synchronous learning occurs in real-time, with all participants interacting at the same time, while asynchronous learning is self-paced and allows participants to engage in the exchange of ideas or information without the dependency of other participants involvement at the same time.

Nowadays, there are many open course learning systems available which would interest the students and help them with their present discussion. Course Recommendation enables the system to better help the students and their learning process. Recommender systems are active information filtering systems that attempt to present to the user information items (film, television, music, books, news, web pages) the user is interested in. These systems add information items to the information flowing towards the user, as opposed to removing information items from the information flow towards the user. Recommender systems typically use collaborative filtering approaches or a combination of the collaborative filtering and content-based filtering approaches, although content-based recommender systems do exist. Our system is specifically designed to address this issue of recommending specific courses and information depending upon the interests of the students and their present search criteria. It uses an artificial-intelligence algorithm to do the recommendation in an efficient manner considering the specific interests of the students and other search patterns.

2. BACKGROUND

The current trending culture of e-learning and online study drives us to this novel idea of combining online study and course recommendation. There are many successful and established systems in the past which provide a unique study system or a good course recommendation but very less ones which provide both the important features on the same platform. Most systems in the past have provided essential features like technical interaction or group study but have not been extended to course recommendation which we think is the most essential part in an online study system. Course recommendation remains one of the most crucial part considering its importance with time and effort. Thus, combining these systems would be beneficial to the users.

3. APPROACH

There are three major tasks in this project. Firstly, collecting and processing data for the recommendation systems. Secondly, using algorithms and special techniques from machine learning to come up with relevant suggestions. Our final goal is to provide a good online study environment for the users tweaking in the aforementioned features for groups to collaborate efficiently.

3.1 Course Data Evaluation

Any online course has some attributes like course name, university name, course date, info, etc. These attributes just can meet students basic need to analyze the course and determine if it is useful for the students. In addition to those attributes, we want to add more detailed information like complexity, students satisfaction, student reviews and related course. All the attributes are important for students to make their choices and they are essential to our recommendation system.

3.2 Recommendation

Our recommendation system should not be a passive one taking certain inputs and then searching database rather it has the ability to act dynamically through past inputs as well. The user may or may not have taken some courses recorded on database, we can use past courses information with user's preference to predict most relative course the user may want to enroll in the future, all the procedure is according to the attributes described above. For example, a nascent computer science student, after finishing data structures course would prefer to know more about programming. Thus, we would recommend him courses teaching C/C++/Java or programming studio or intro to algorithm, etc.

3.3 Methodology

This project employs a basic online system developed primarily on PHP and JavaScript. Some of the steps to be taken are:

- Feasibility Study:
  Understanding and identifying of existing course recommendation and study systems with an associated study of the features that can be incorporated in such a system.
- Analysis:
  Proper analysis of the features that are useful and would support the users.
- Coding:
  The system is implemented on a 2-tier basic architecture model using PHP language. The 3 layers of the application are as follows:
  1. HTML, CSS and JAVASCRIPT - Front End
  2. PHP - Middle end
  3. MySQL(Php MyAdmin) - Back-end

3.4 Software Usage and System Requirements

- Hardware Requirements
  - 512 MB RAM or more
  - Windows 7 or upgraded versions
- Programming Languages and Environment
  - PHP (Development kit)
  - WebMatrix (IDE)
  - MySQL (Database)
- Backup Media
  - Hard Disk

This system is developed using Windows-8 as system software and is tested to be executed on all the operating systems given above.

4. IMPLEMENTATION

Most of the current online systems are based on the Internet, hence our system needs to powerful in a web-context and next be a cross-platform application. In addition, we need a reliable database to store our course information, so that we combine the use of PHP with MySQL to build the system to have extreme coherence and compatibility. A login system is very important on server like ours to distinguish different groups, we need to identify every user and record their information, so that login user is necessary. PHP also works efficiently with MySQL database, which is very fast, reliable and easy to use.

In order to deal with the data between user and database, specially designed algorithms will be used making the information exchange robust at the backend. Also, HTML and CSS along with some implementation of JavaScript will design the user-interface to be attractive and user-friendly. Bootstap has been used to maintain consistency across all the clients. Lastly, the credentials of the users logged in will be stored using the database connectivity.

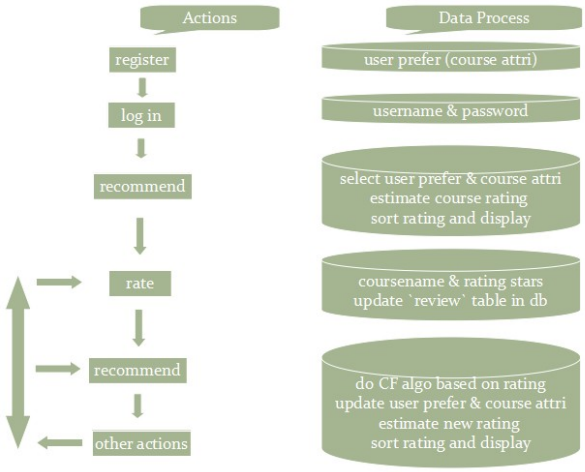

Figure 1: Process flow

4.1 Overview

The system is a web-application built mainly using PHP for the functionality and MySQL to store the application information for retrieval and injection. Also, xampp has

been used to incorporate the server functionality and PHP-MyAdmin features available for the development and testing. The database has been implemented to store the information of the user, course and preference. The recommendation algorithms uses the database to retrieve information based on the user preference and rating given to a specific course.

Figure 2: Login-In system

4.2 Study Room

One of the major goals of this project was to provide a good study system in a distributed environment. Thus we have a built-in chat system for the users to exchange textual information and conceptual data as they study. Also, the internal frames within the web application allows the users to stay connected during the study and also exchange information among them regarding the course that they are presently studying in a distributed environment.

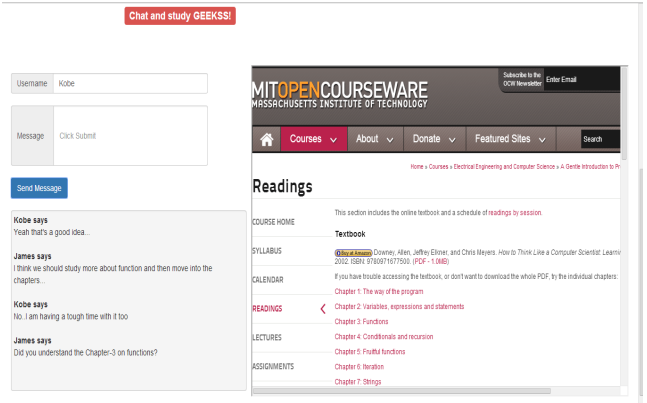

Figure 3: Study Room

The chat system functionality has been developed with PHP, JavaScript. The CHAT database has been used as shown in the figure to allow the user to retrieve any links or important information from their history. This is very important in order to promote collaboration and enhance user-user communication.

4.3 Database Flow

For this system, we need five tables to store user and course information and their relations. Each user can enroll in many courses, once the user has finished certain course

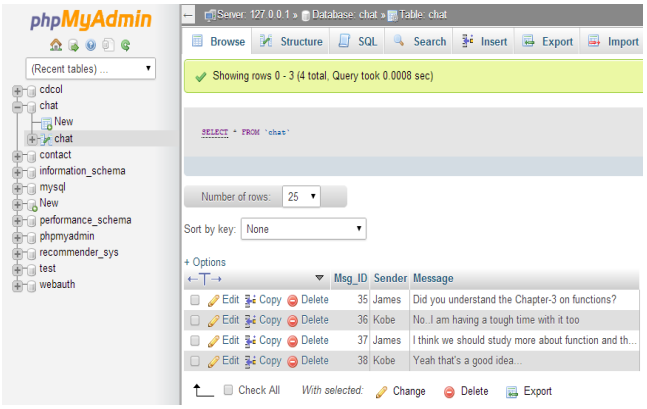

Figure 4: CHAT database

he will rate it and provide review. All the user preference and course attribute are stored accordingly as shown in the figure, now it is easy for the system to predict users interest.

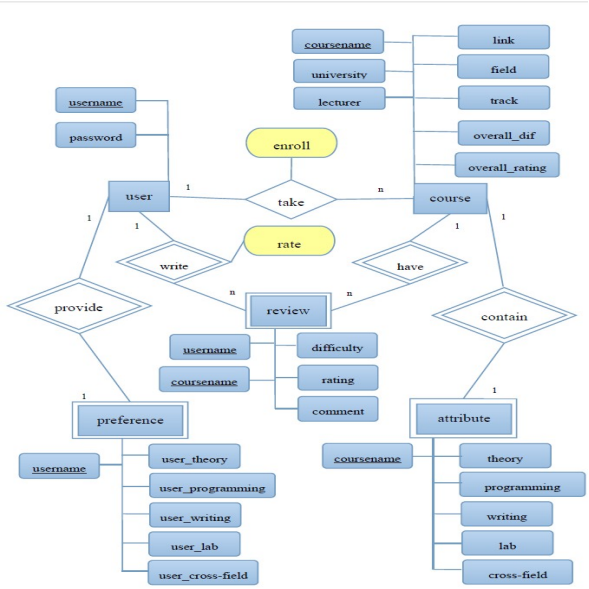

Figure 5: Relational database structure

5. COURSE RECOMMENDATION

5.1 Based on user preference

The Online Recommendation system will work on two main algorithms for the recommendation process. In the first algorithm, the system recommends courses based on the user preference and their need for the hour. The user enters his choice considering his skill on a range of areas and then decides which course to take up from the recommended ones. The course information has already been loaded from different online opencourseware websites. The Figure.4 shows the areas taken into consideration and how recommendation will happen based on the user preference and their experience in these areas.

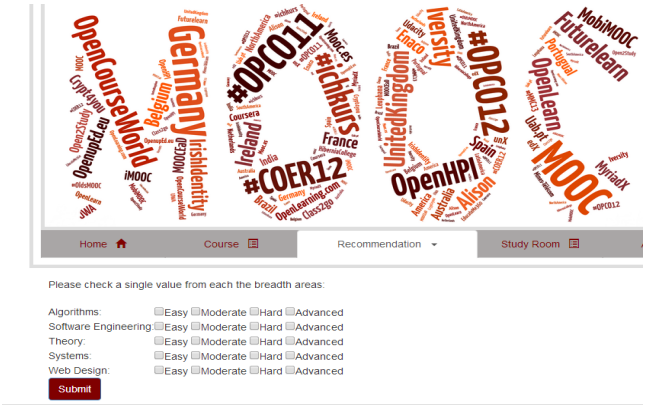

Figure 6: Recommendation based on user preference

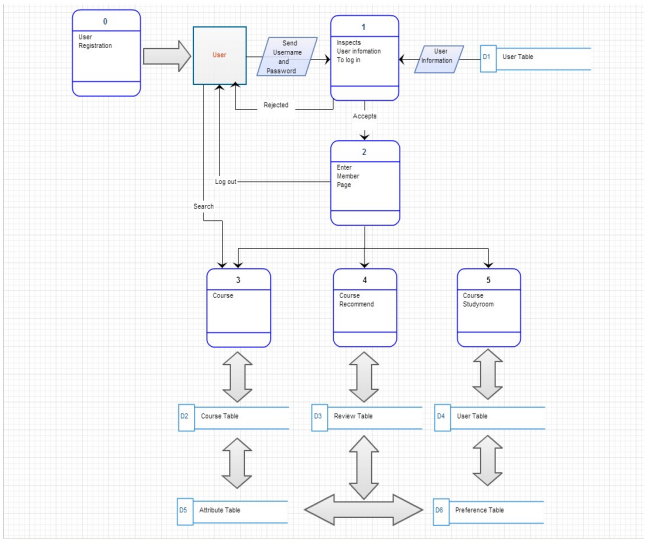

Figure 7: System dataflow

5.2 Based on Collaborative Filtering

Collaborative filtering (CF) is a technique used by some recommender systems. Collaborative filtering has two senses, a narrow one and a more general one.In general, collaborative filtering is the process of filtering for information or patterns using techniques involving collaboration among multiple agents, viewpoints, data sources, etc. Applications of collaborative filtering typically involve very large data sets.

Collaborative filtering methods have been applied to many different kinds of data including: sensing and monitoring data, such as in mineral exploration, environmental sensing over large areas or multiple sensors; financial data, such as financial service institutions that integrate many financial sources; or in electronic commerce and web applications where the focus is on user data, etc. The remainder of this discussion focuses on collaborative filtering for user data, although some of the methods and approaches may apply to the other major applications as well.

In the newer, narrower sense, collaborative filtering is a method of making automatic predictions (filtering) about the interests of a user by collecting preferences or taste information from many users (collaborating). The underlying assumption of the collaborative filtering approach is that if a person A has the same opinion as a person B on an issue, A is more likely to have B's opinion on a different issue x than to have the opinion on x of a person chosen randomly. For example, a collaborative filtering recommendation system for MOOC course tastes could make predictions about which course a user would like given a partial list of that user's preference(likes or not likes). Note that these predictions are specific to the user, but use information gleaned from many users. This differs from the simpler approach of giving an average (non-specific) score for each item of interest, for example based on its number of votes and its value of likes. A simple example is shown below.

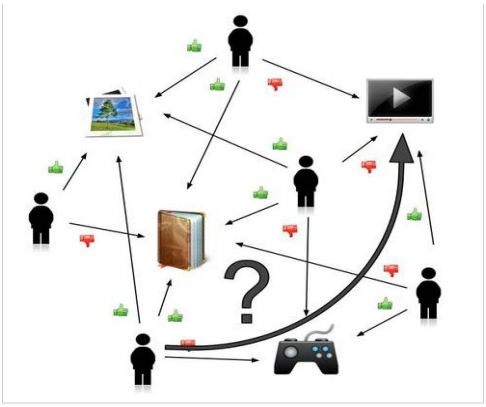

Figure 8: Collaborative Filtering

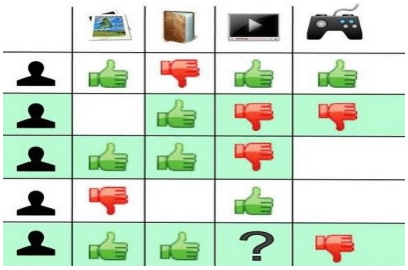

Figure 9: Collaborative Filtering-1

For a group of people who have their own preferences, we want to predict a certain user over a certain item in the list. By using collaborative filtering, we can come up with a simple view about the users preference over this item.

5.3 Our Recommendation Algorithm

5.3.1 Content based

A content-based recommendation system recommends an item to a user based upon a description of the item and a profile of the user's interests. Content-based recommendation systems may be used in a variety of domains ranging from recommending web pages, news articles, restaurants, television programs, and items for sale. Although the details of various systems differ, content-based recommendation systems share in common a means for describing the items that may be recommended, a means for creating a profile of the user that describes the types of items the user

likes, and means of comparing items to the user profile to determine what to recommend. The profile is often created and updated automatically in response to feedback on the desirability of items that have been presented to the user.

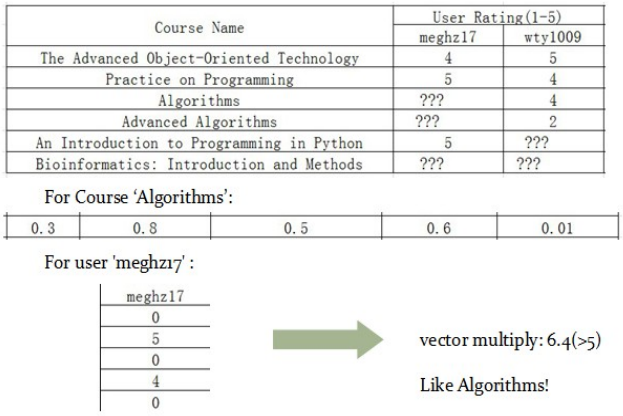

Figure 10: Working of the algorithm

In the figure above, we can see that user 'meghz17' has taken three courses with rating 4 for OOT, 5 for Practice on Programming and 5 on Introduction to Python, while user 'wty1009' has taken four courses with rating 5, 4, 4 and 2. Those ratings are based on a 1 to 5 scores scale, which shows how much a user likes a certain course. We begin to find a certain course 'Algorithm' with its feature vector, also a certain user 'meghz17' with his attribute vector. Since all the features are shared between course and user so that the dimension of these two vectors are the same. Then we do a vector multiplication with course feature and the transpose of user attribute, which comes a relatively score of likelihood. In this case, we get a score of 6.4, which means that user 'meghz17' is likely to choose 'Algorithm' based on his interests.

	User Preference(0-5)	
	meghz17	wty1009
Theory	0	5
Programming	5	4
Reading/Writing	0	0
Lab/Project	4	5
Cross Field	0	0

Figure 11: Example course rating

Using the same method, we come up with the final result of the scenario. For other courses that the user has not taken, we get a predicted value for them. For user 'meghz17' we may recommend him with 'Algorithm'(6.4 score) rather than 'Advanced Algorithm'(3.4 score), while probably not to recommend 'Bio-informatics'(<1 score). For user 'wty1009', we may recommend him with both 'Introduction to Python' (>5 score) and 'Bio-informatics'(4.95 score).

5.3.2 *Algorithm Improvement*

Content-based recommendation works really good but we find that manually inputting all the course features and user

Course Name	Course Attributes(0-1)				
	Theory	Programming	Reading/Writing	Lab/Project	Cross Field
The Advanced Object-Oriented Technology	0.2	0.8	0.2	0.99	0.1
Practice on Programming	0.2	0.9	0.3	0.9	0.1
Algorithms	0.3	0.8	0.5	0.6	0.01
Advanced Algorithms	0.8	0.2	0.8	0.6	0.01
An Introduction to Programming in Python	0.2	0.99	0.3	0.99	0.01
Bioinformatics: Introduction and Methods	0.9	0.1	0.8	0.01	0.99

Figure 12: Example course table

attributes is time consuming and not the intuitive idea of our system. So we implement collaborative filtering in a machine learning way, that is to use algorithm let the system learn those features automatically. In detail, given course attributes can estimate user preference. Having the cost functions based on given features, we use linear model minus square part in both cost functions actually dealing with the same feature of vector multiplication, so it gives us a chance to update both course feature and user attribute at the same time.

First of all, we initialize course attribute and user preference to small random values. Minimize cost function with their values, using gradient descent algorithm to update these values. At last, for a user with certain preference and course attribute, predict the rating.

5.3.3 *Evaluation of Algorithm*

For user 'wty1009' with preference vector: (0.4, 0.3, 0.5, 0.4, 0.5), we initially recommend him all the courses based on initial preference vector. But after the user study and rate a certain course, both the course features and user attributes will be updated to a optimal value to fit in our learning model. For example, after 'wty1009' rated 5 stars for 'Algorithm', which according to our system achieves highest estimated rating among all the courses, and then he also rated 5 stars for 'An Introduction to Interactive Programming in Python', which achieves second highest estimated rating in the system. But after the user rated 5 stars for 'Practice on Programming', besides the courses he has already taken, which achieves the highest estimated rating now, the estimated rating list have some changes. The previous order of the recommendation is:

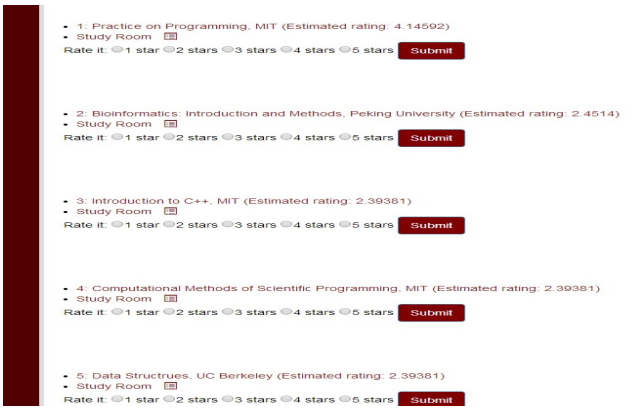

Figure 13: Before Recommendation

- Previous order
  1. Bioinformatics
  2. Introduction to C++
  3. Computational Methods of Scientific Programming
  4. Data Structures
- Previous order
  1. Computational Methods of Scientific Programming
  2. Data Structures
  3. Bioinformatics
  4. Introduction to C++

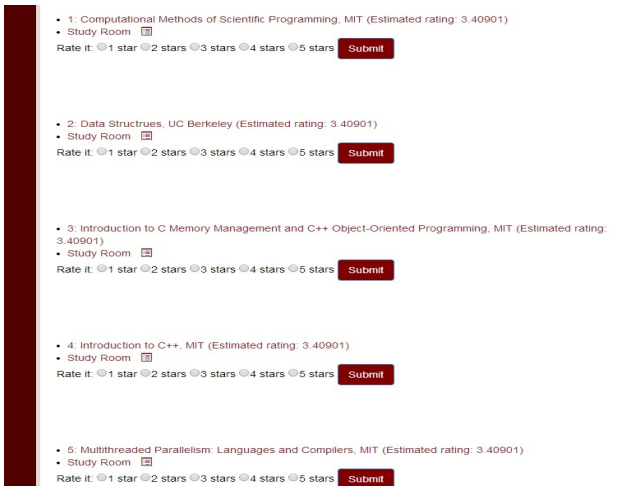

Figure 14: After Recommendation

6. USER STUDY

One of the most important task of our project was to evaluate our system with users physically using it. This would give us a fair idea on specific area that needs potential development. Also, many design issues and flaws would be exposed could be eliminated by conducting a user study of this system with students who would be the potential users of our system in the near future.

6.1 Participants

We had 16 participants in our user study who carefully evaluated our system for various technical flaws and to provided their opinions about the system. They had no external influence explaining them about the system as to create an environment as if they were using one of those trending course recommendation system or study room out there. Most of the participants were college students and colleagues who were studying at the Texas A&M University-College Station. All of them had prior experience with online courses and were given a demo about the various features of the system.

6.2 Evaluation

The user study was conducted by asking the candidate several questions related to the system and how they felt about each of these incorporated feature. Some of the questions were:

6.2.1 *Did you use a similar system before?*

This question was initially posted to know if the user had prior experience using any system of similar kind. From this we could find valuable data as they would relate our system with the ones they used before. Our study shows that 56 percent of the people among the 16 participants had used a similar kind of system before.

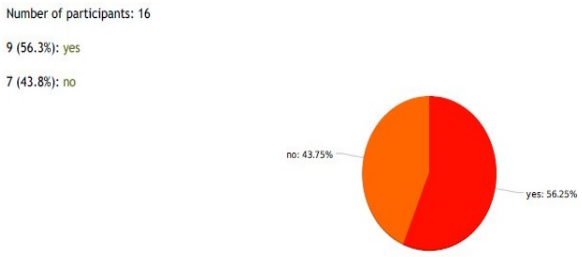

Figure 15: Percent of experienced users

6.2.2 *How many online courses in an year?*

We wanted our system to be evaluated by people who had good knowledge with online courses and experience hence we posted this question to know what was their level and interest in online systems. This would particularly guarantee us users who had fair idea about our system and thus the result set and suggestions would be accurate for further development.
From the pie chart below we observe that many users had taken at least 3 online courses in an year and others took more than 3 too. Many had commented that they did not complete the course fully which may due to several reasons; one being the course content but that was not taken into consideration as we only wanted to know about the basic functionality of the system, recommendation algorithm and other design issues we could further develop.

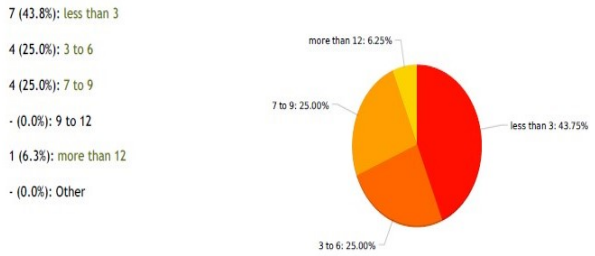

Figure 16: Proficiency with online courses

6.2.3 *Satisfaction with current features?*

In order to know how the users felt about the various features and measure the level of satisfaction we posted this in the user study. We observed that among the 16 participants 12 of them liked the idea of a collaborated study room with course recommendation. Also, they liked the chat system which gave them chance to communicate with their peers. Some of them complained about the basic user interface and thus that remains as a challenge to work and consider the design interface carefully in the future.

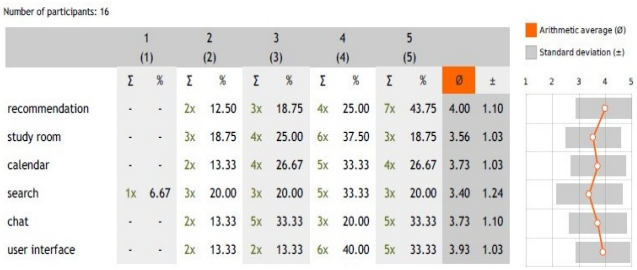

Figure 17: Level of satisfaction

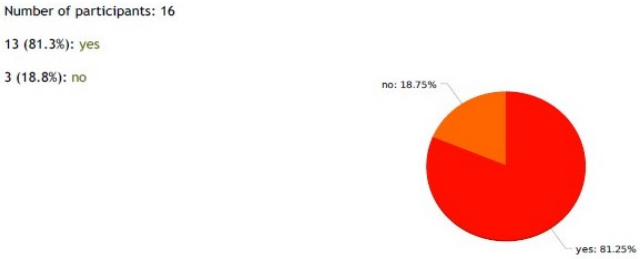

Figure 19: Percent of user satisfied

We also observed what their favorite feature among the current ones so that future work could be concentrated on developing these features. As most of them were college students the mental behavior and likes of this age group would be the same. From the results obtained study room was the applauded by 12 amongst the 16, and the next was the the recomendation followed by chat, calendar and search. This gave us a fair idea on which feature was the most popular and could be developed to attract more and provide better assistance.

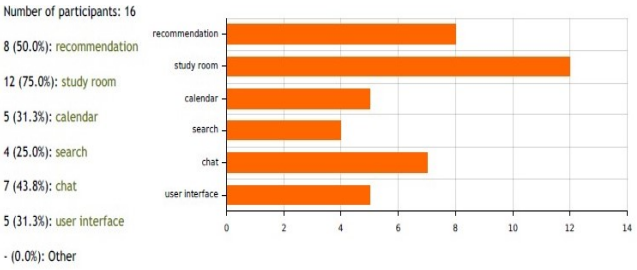

Figure 18: Best feature according to the study

6.2.4 Would you like to use our system again?

Finally, we wanted how the users felt about using the system again to gauge how popular the system would be if deployed. To our surprise, 13 amongst the 16 participant said they were interested in our system and would return to a more developed version when deployed. This shows the currently growing interest towards the online course system and migration from the traditional class room teaching towards e-learning. There were users who did not want to use this kind of system in the future and were more interested to the class room teaching. This potentially maybe due to the flaws in the system or their inclination towards the conventional system of study.

6.3 Design lessons

We have made many essential observations from the study that people preferred a simple interface with robust features. Although, most of the students were technologically driven they preferred a simple interface and did not want the ordeal of handling a complex system. This is mainly due to the fact that systems main functionality is to provide the user a good learning experience and not perplex them with its complexity. We have acquired the user interests to further develop the system accordingly. It has been noticed that many users i.e the students have turned to the e-learning and systems like ours will be popular in the near future if carefully developed considering their needs and improving them accordingly.

7. CONCLUSION

This paper introduces the Online Study and Recommendation system. The basic system has been presented in this paper and design issues have been exploited by conducting a user study. Our approach in building such a system is mainly directed towards the computer supported cooperative work considering the individual's behavior working in a group. Thus, the algorithm developed will also ensure such an activity at an intense level. Also, the languages used and API required to build the Online Study and Recommendation system have been mentioned within the paper. Through our design process, we thought critically about how the unique social benefits of our system could benefit this population and crafted an application to support the needs of this group.

Secondly, we conducted an extensive user study in the given time frame to gauge our systems capabilities and analyze the basic features of the system in a real environment. We organized our study around three themes: analysis, capabilities of system interaction and recommendation. As expected we received valuable information for the development and direction to further our research in the area. This information and ideas will prompt designers to think differently about the way that system environments can be designed in order to support a wide variety of social interactions and communication amongst the users.

Based on our study, a system which is simple and has important features is more likely to succed than the one which is complex and hard to deal. with a higher level of privacy would succeed on a longer run and thus our system would be a closed system. However, it would be open to a group of individuals with common login credentials.

8. FUTURE SCOPE

Our research on the current systems show that not many have the features discussed earlier in the paper. Thus, the Online Study system will be a widely acceptable web system as it has a broad scope of development in the future. Most other famous technical features will be compatible and can be incorporated into it. For instance, the news recommendation would interest many around the world as it would keep them updated with the current affairs and research trends. Many synchronous features such as such as instant messaging and asynchronous features like the message boards can be implemented along with the system.

Moreover, the recommendation algorithm can be extended to a new set of areas other than opencourseware.

9. ACKNOWLEDGEMENTS

We would like to thank Dr. Frank Shipman for this wonderful opportunity and challenging task. It has been a great learning curve dwelling into the area of computer supported co-operative work. The information and support provided throughout the course of this project is most appreciated.

10. REFERENCES

1. Schroeder, R. (2012). Emerging open online distance education environment. Continuing Higher Education Review, 76 90-99

2. Tschofen, C., & Mackness, J. (2012). Connectivism and dimensions of individual experience. International Review Of Research In Open & Distance Learning, 13(1), 124-143.

3. Schroeder, R. (2012). Emerging open online distance education environment. Continuing Higher Education Review, 76 90-99.

4. Fini, A. (2009). The technological dimension of a Massive Open Online Course: The case of the CCK08 course tools. International Review Of Research In Open And Distance Learning, 10(5), 1-26

5. MOOCs: A Systematic Study of the Published Literature 2008-2012 Tharindu Rekha Liyanagunawardena1, Andrew Alexandar Adams2, and Shirley Ann Williams1

6. Collaborative Filtering-http://en.wikipedia.org/wiki/Collaborative_filtering