Designing an Intuitive Course Recommender System

Rushikesh S Subhedar
University of North Carolina at Charlotte, NC, 28223
rsubheda@uncc.edu

Kritika Dutta
University of North Carolina at Charlotte, NC, 28223
kdutt1@uncc.edu

Sepehr Sabeti
University of North Carolina at Charlotte, NC, 28223
ssabeti@uncc.edu

ABSTRACT

In this paper, we are suggested a user-friendly and conversational-based course recommendation system for university students. The proposed course recommended system will provide a course recommendation to the students based on students' degree, concentration, interest, subject competency level, industry trends, and career prospects to make the appropriate course selection for their more industry-oriented readiness.

Keywords

Recommender system; course registration system; course recommender system; KNN; k-nearest neighbors' algorithm; conversational interface; Mobile App; Mobile-based course recommendations; smart education

1. INTRODUCTION

Most of the students get a college education to better their chances of getting a good job. The primary student needs to select a set of courses/classes for each college semester to complete their mandatory courses. The selection of the right course as per degree, concentration is essential.

However, there are many indirect parameters which student attempt to evaluate and predict before enrolling for any course—for example, the future industry trends, competency support, career prospects, job opportunities, etc. Unfortunately, most of this information is not readily available and not integrated with any educational course registration system. Students need to rely on other information sources only to proceed with the right course selection.

There is a real need for a next-generation course recommender system for students to overcome this issue. The objective of this study work is to design and develop a modern and intelligent course registration system that will allow users to use the course registration in a user-friendly way and to get a personalized course recommendation. The proposed course registration system would also consider students' competency level, real-time industry trends, and need to make the appropriate course recommendation for their more industry-oriented readiness.

To personalize information, recommender systems can be used either to recognize a similar user or to identify particular objects of the user's interest. recommender systems prioritized the information, related to items and provides user with meaningful recommendations as per their interest. Recommender systems are subset of information filtering concept having immense potential to help users in creating personal learning environment by identifying the most relevant and Interesting items from a large database then recommend it the user based on preferences and interest.

2. RELATED WORK

This section shows some of the relevant work on various approaches to recommender systems. Recommender systems use different information retrieving techniques to find and recommend items of interest to their users. Therefore, if a recommender system can recognize the intent and requirements that a user expresses in the form of queries, it can generate more valid recommendations.

In Google news, 38% of the total views result from recommendations; similarly, 60% of the rented movies from Netflix come from recommendations. More than that, Amazon's sales percentage due to recommendations is 35%. Therefore, recommendation systems are considered an impending factor in business nowadays. But in the education sector, learners are still being provided with static and predefined patterns of learning courses, tasks, materials, objects.

However, learners differ in learning objectives; interests, objectives, needs, skills, and personalities. Taking all these differences into account, it has become essential for personalizing learning items for individual learners using recommendation systems [1].

2.1 Recommendation System in Education

Recommender systems for educational systems are essential because they provide personalized information in learners' interest by providing relevant and customized information. Artificial Intelligence techniques in this respect might be used to develop and imitating human decision-making and reasoning processes to minimize uncertainty for effective learning to ensure lifelong learning mechanisms [2].

Recommendation systems strongly depend on the domain to operate upon, and it's not easy to take recommendation from one system and transfer it to another domain. Therefore, the challenge for recommender systems is to understand the learner's interest and the purpose of the domain better [3].

An ontology-based model was proposed for E-learning personalization to recommend learning objects by refereeing the past preference history of learners. This system also suffers from cold start problems like traditional systems and is limited to learning objects only [4].

A hybrid recommendation system was built for course recommendations with professor and student information datasets. Since this recommender system's base is collaborative, it uses a filter of professor's ratings in course recommendations. It proves that the quality filter in terms of professor's ratings does not interfere with the predictions of the proposed recommender system [5].

New E-learning systems appear to ensure personalization in terms of content, but the courses and such issues can be overcome by applying recommender systems to E-learning systems [6].

3. SYSTEM DESCRIPTION

This system is designed as a commercial application of a recommender system in education. The system integrates a devised early user research, an early implementation of a conversational recommender system, and a follow-up usability test. Below, each component of the system is defined in greater detail.

3.1 Recommendation algorithm flow and system model

We are using K Nearest Neighbor (KNN) as the main recommender for the proposed pilot course recommender system. With the given algorithm, the system probe for three questions to the students about the courses they are interested in taking, the difficulty of the course, and the programming requirements and then runs the KNN using Mismatch distance to find the closest matches to students' inputs. Then, the model provides the four closest options among the offered courses to students' inputs.

Please note that the conversational chatbot questions can be replaced based on various dataset parameters. With the current pilot study, we are just exploring the capabilities of the KNN to check the course recommendations' qualities.

3.2 Conversation interface

We decided to use conversational interface-based interaction for course recommendations to engage the student and make the overall user interaction more engaging.

Conversational interface is all about the user experience and satisfaction. Conversational interface enables users to interact with a machine through a combination of a natural language with buttons, menus, images, and other graphical UI elements. There are two basic types of conversational interfaces. There are voice assistants, which you talk to, and there are chatbots, which you type to.

3.2.1 Reactive and proactive conversations

We also decided to use both reactive (students can initiate conversation based on the need) and proactive (system can initiate the conversation based any specific instance; for example, if student spend more than 30 seconds on the application home page. Etc.) type of conversational features for the proposed university course registration system.

3.3 GUI design

For this study, we designed and developed the mobile app-based course recommender system. The primary reason for Mobile interface is that mobile-friendly interface solutions are very popular and influential within the student's community. As per the recent study, two-thirds of online students do at least some of their coursework on their phones [7].

3.3.1 Information architecture (IA)

For the proposed mobile interface allows students to navigate between minimum screens to reach the intended course recommendations.

Students can launch the App, and post-authentication, navigate to the app home screen. The home screen would allow students to start the course-related conversation reactively or proactively initiate the conversational screen. Students can interact with the chatbot and answer's the questions naturally. If the student's input satisfies any course(s) suggestions, the App will end the conversation and navigate the recommended course(s) list.

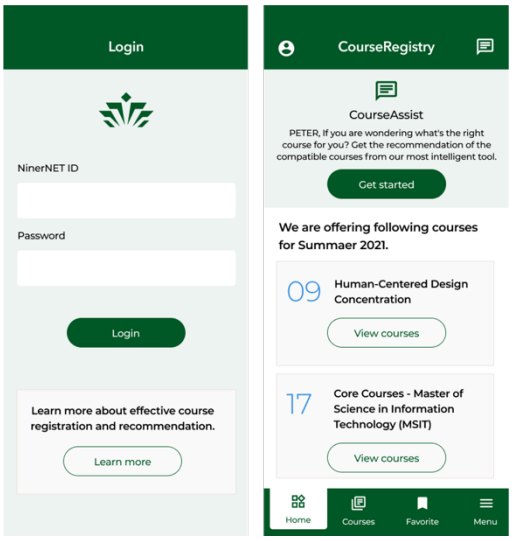

Figure 1: The proposed App login and dashboard

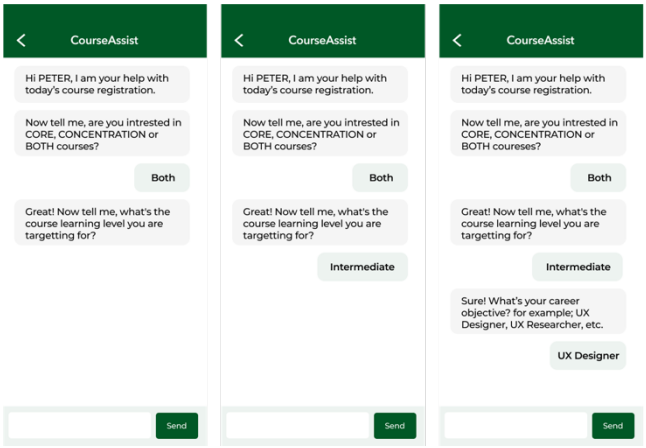

Figure 2: The proposed App conversational interface and interaction.

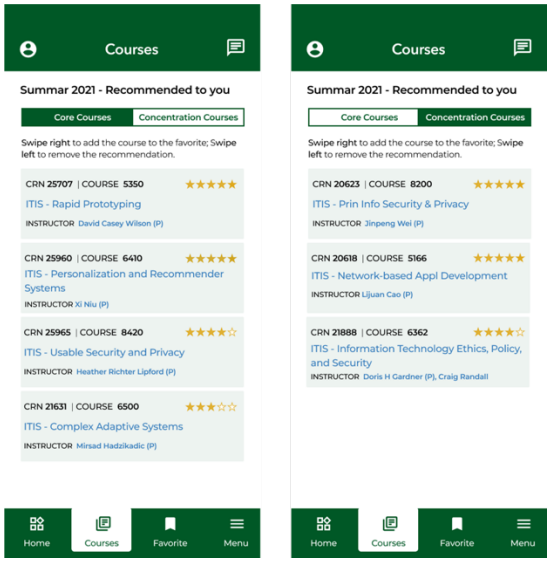

Figure 3: The proposed App course recommendations.

3.4 Dataset

While designing the pilot system, our first challenge was finding the correct data that could potentially be used, interpret and interact with the proposed system. By analyzing our peer student's combination of courses and the performance in these courses, we have built some course combinations that will eventually inform the user about which course will have more advantages considering various individual student-specific conditions.

The sample dataset created shows many characteristics of the student's course-related information. We captured the following data elements/columns for each course while building our dataset.

We have manually created 100 courses dataset with the following elements of characteristic. Please note that most of the below information parameters are not shared by the university or not available on the UNCC website. We have used some of the parameters with the predicted values (based on discussion with a small set of students.

Course_number	Course_name	Course_type
Course_credit	Course_offering	Course_rating
Course_rated_by	Enrolled_student_count	Programming_skills
Course_difficulty	Skills description	carrer_mapping
percentage_of_new_career_starts	instructors	course_start

Table 1: Sample dataset for course recommender

4. TECHNOLOGIES

The proposed pilot recommender system is written in Android. Android is one of the most popular and widely used mobile operating system in the world. It is based on a modified version of the Linux kernel and other open-source software, designed primarily for touchscreen mobile devices such as smartphones and tablets.

Also, we are using Firebase (Firebase is Google's mobile application development platform that helps build, improve, the android mobile app.) user authentication and data storage. Users can create and log in through firebase services. The courses offered each semester are also stored in this service.

5. EVALUATION STRATEGIES

Before we evaluate our recommender system it is important to take a look at what our recommendation system is trying to achieve. As we briefly mention in the description of the system that it focuses on finding viable suggestions for the choosing the right course based on student's history. Understanding the context in which the recommender system will be used is important when deciding how to evaluate the recommender system.

5.1 Initial research - Survey Analysis

We conducted an online survey to understand the student's current course registration thought-process, expectations, need, etc., from the university course registration system.

We collected 11 participant's inputs (UNCC students) for the initial survey. All participant's inputs are considered during the

conceptualization and design stages. The survey helped understand-

1. How students select the course(s)?
2. What are the most important things students would like to consider while selecting the course?
3. What they expect from the course registration system?
4. Students' current experience with the course registration system,
5. Future course application expectations, etc.

5.1.1 Survey findings

Some of the findings from the initial research survey:

- 45.5% of the students found UNCC's current course registration system confusing.
- 63.63% students believed course rating information is essential while registering for the course.
- 90.3% of them would like to get course recommendation aligned with their degree and concentration requirement.

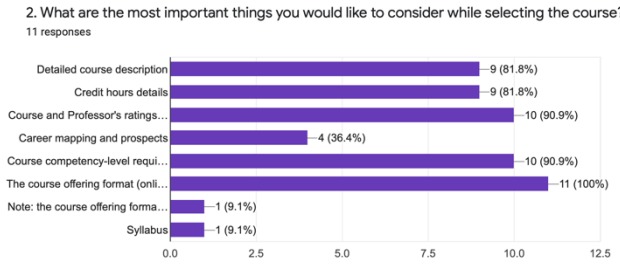

Figure 4: Survey findings – The most important things students would like to consider while selecting the course.

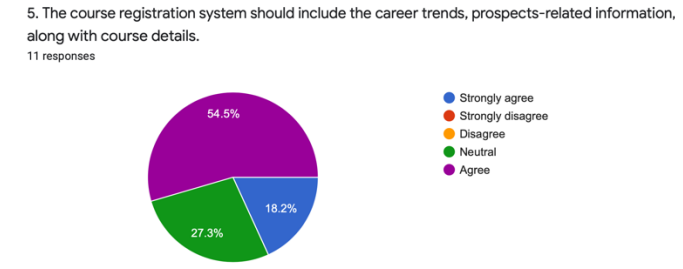

Figure 5: Survey findings – The course registration system should include the career trends and prospect information.

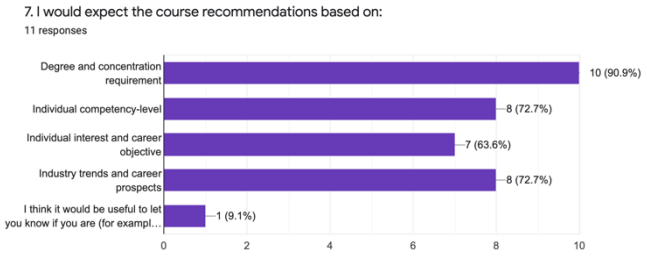

Figure 6: Survey findings – Student's expectations about course recommendations.

5.2 Usability evaluation

We have performed usability evaluation that focuses on how well users can learn and use a product to achieve their goals. We evaluate the effectiveness, efficiency, and the overall satisfaction of the users. Our main points for usability evaluation are –

1. If the interface is intuitive and has a correct architecture flow and navigation.
2. Whether the system is easy understand for the user and if he can accomplish basic tasks.
3. How fast can user accomplish the task.
4. If user remembers all the functions to effectively use it in future.
5. How often user is making errors while using the system and if those errors are serious or users can recover from it.

We conducted user studies user interviews, user observation while interacting with the proposed prototype and Likert-scale based evaluation with five students at the University of Charlotte to understand their interaction experience with the proposed course recommendation system.

5.2.1 User Interview

We included a set of pre-tasks, user interview open-ended questions for students.

- How often do they face difficulty in finding a course that interests them?
- How satisfied are they with the current course registration system?
- How much time do they spend in figuring out the courses?
- Any positive and negative experiences with the current course registration system?
- How do they define the course complexity? etc.

5.2.2 *User Observation*

Then we had some tasks performed by users in our app. The Primary objective of user observation was to understand how students interact with the proposed interface, observe their task, task completion, failures, etc. Please note that we did not evaluate the student's task efficiency (task completion time) as this was not the primary objective.

Students were first asked to log in with predefined login information). Then the application promoted students to provide input one by one while interacting with the conversational bot about their course type, course difficulty, and programming skill level. Every time users provided wrong input to the system, the chatbot asked them to correct their input details.

5.2.3 *Overall experience*

Finally, we asked users some post-task questions and evaluated their overall experience with the proposed course recommendation solution. Post-task questions were captured on the Likert-scale.

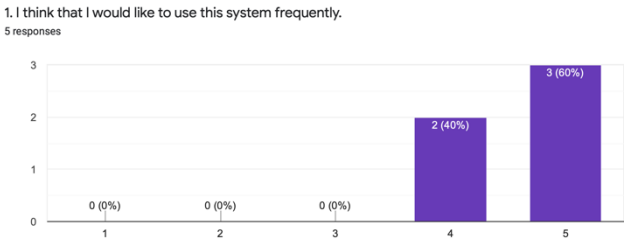

Figure 7: Overall Experience on the Likert Scale – Student's willingness the use the proposed system (1 – I don't want to use 5 – I will surely use)

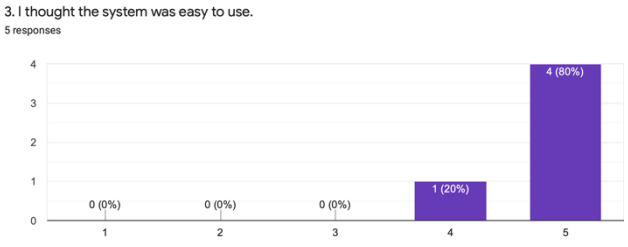

Figure 8: Overall Experience on the Likert Scale – User friendliness of the proposed design (1 – I don't agree 5 – I agree)

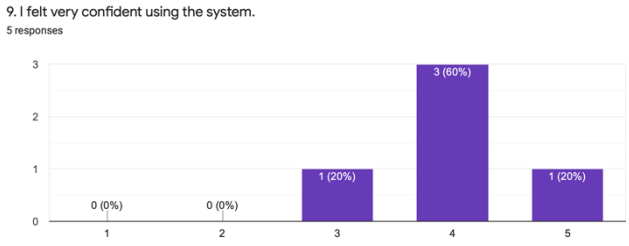

Figure 9: Overall Experience on the Likert Scale – students confidence level after using the proposed system (Scale - 1 – I don't agree 5 – I felt very confident)

5.2.4 *Summary of findings*

Some of the findings of our evaluation proves that.

- Most students said that they would like to use the proposed course recommender system frequently.
- The course registration system is easy to use, not complex, and students can use the system without any technical support.
- Most students feel their peers would learn to use this system very quickly.

6. CONCLUSION AND FUTURE WORK

We presented a pilot version of the course recommender system that easily allows students to select the course based on their personal preferences and interest. The proposed mobile App helps students with their decisions to choose the courses during the course registration process at the beginning of each semester.

Students always prefer to select highly rated course(s) by past students. Students also like to select easier, competency matching, and career prospect courses defined within their courseware. The proposed App would work based on students' personalized parameters or conditions to deliver a list of recommended courses.

The future goal for this project is that integrating the course recommender system within the university application cluster. Given that students can use university login credentials to log in to the course registration and recommendation application. The proposed course recommendation system can be integrated with students' degree work to get valuable inputs like student's grades, current, past course registration, GPA/score, degree, and concentration details to evaluate the actual competency. Once the recommender system is developed, it can easily be extended to the entire university and integrated into the UNCC Charlotte website.

Since in this study we evaluated the effectiveness of the mobile based recommendation system. Next, we plan to build our web-based based application using python and its API to build our recommender system. The API we are going to use is Chatterbot [8]. Chatterbot comes with a data utility module that can be used to

train the chatbots. This API helps in creating software that engages in conversation.

Our application includes KNN model which is built entirely from the data given to it. This algorithm is used by companies to recommend different movies, to watch or books to buy. In order to implement KNN the first step is to transform data points into feature vectors, or their mathematical value.

The algorithm then works by finding the distance between the mathematical value of these data points. One way is to find the distance is by calculating the Euclidean distance. Then KNN finds the similarity between the two points and classifies it based on which points share the highest probabilities.

$$d(\mathbf{p}, \mathbf{q}) = d(\mathbf{q}, \mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$$
$$= \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}.$$

Figure- 10 Euclidean distance formula measure the similarity between two data points.

Also, for future work, we recommend getting various additional input from the university like student ratings for courses and professors, descriptive course description, prerequisite skills details for each class, industry trends, career prospect mapping for each course, prerequisite, dependent course, etc. details. These information parameters would make the course recommendation more intelligent and student-centric.

7. REFERENCES

[1] Lim. (2015) “Recommendation System using R” . CHAOS Book

[2] A.Khalid, H.H. Daniyal, and A.I.Ghadah. (2017) “ A Survey Of Artificial Intelligence Techniques Employed For Adaptive Educational Systems Within E-Learning Platforms.” JAISCR, 7(1): 47-64.

[3] K.M.Aleksandra, B.Vesin, I.Mirjana, B.Zoran, and C.Lakhmi, Recommender Systems in E-Learning Environments :E-Learning Systems : Intelligent Techniques for Personalization. Springer International Publishin, 112, 51-75.

[4] C.Y Huan, R.C.Chen, and L.S.Chen. (2013) “Course-Recommendation System Based On Ontology.” *IEEE International Conference on* Machine Learning and Cybernetics, Tianjin, 14-17.

[5] C.Pei-Chann, L.Cheng-Hui, C.Meng-Hui, A Hybrid Course Recommendation System by integrating Collaborative Filtering and Artificial Immune Systems. Algorithms 2016, 9, 47.

[6] S. Benhamdi, B.Abdesselam, C.Raja, ‘Personalized recommender system for e-learning environment’, Educational Information Technology, Springer (2016)

[7] Survey of 1500 students conducted by Learning House: https://campustechnology.com/articles/2018/06/19/two-thirds-of-online-students-do-some-coursework-on-a-mobile-device.aspx?admgarea=topic.mobilecomputing

[8] https://www.edureka.co/blog/how-to-make-a-chatbot-in-python/