WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Elective Recommendation System

[1]Pokala Siva mani Reddy, [2]Valipireddy Venkata Sainath Reddy, [3]Pullalacheruvu Mahitha, [4]Meda Sai Santosh, [5]Shaik Mohammed Saif

[1,2,3,4,5] Dept. Of SOCSE, Presidency University, bangalore-560064

[1]Pokalasiva12@gmail.com, [2]v.sainathreddy27@gmail.com, [3]pmahithareddy730@gmail.com, [4]Meda.saisantosh123@gmail.com,

[5] saifshaik1609@gmail.com

ABSTRACT –

Selecting elective courses can be a significant challenge for students at Presidency University, often leading to dissatisfaction and inefficiencies in course allocation. Common problems include students not receiving their preferred electives, disconnection from peer groups, and arbitrary course allocation to balance class sizes. This paper presents an AI-driven elective recommendation system that leverages Artificial Intelligence and Machine Learning to enhance elective selection and allocation processes. The system employs a hybrid recommendation model combining collaborative and content-based filtering, analyzing factors such as academic performance, interests, peer dynamics, course capacities, and institutional policies. A user-friendly interface enables students to make informed decisions, while administrators benefit from tools to optimize allocation and maintain fairness. A pilot deployment at Presidency University highlights improvements in student satisfaction and administrative efficiency, demonstrating the scalability of the system for broader academic applications.The system architecture of such a system is built to be comprehensive, easy-to-use, and effective so the users do not have to use multiple systems for the same thing, and for healthcare in this case, one is enough. This project focuses on the gap in the market for combined health services which is technologically savvy and does not compromise on the patient experience.

Keywords: **Elective recommendation, academic planning, Artificial Intelligence, Machine Learning, hybrid recommendation system, course allocation, student satisfaction.**

I. Introduction

Selecting electives is a pivotal aspect of a student's academic journey, enabling exploration across disciplines, diversification of knowledge, and alignment with personal interests and career aspirations. However, the increasing variety of elective courses has made this process increasingly complex and often overwhelming for students. Challenges such as identifying electives that align with interests, career goals, and academic strengths, coupled with constraints like course availability and timetable conflicts, often hinder optimal decision-making.

This process has become very complicated and frustrating for the students themselves as well because the optional courses are on the rise also. Challenges would include optimizing choice over electives, relating to both interests and career goals and one's academic strength while avoiding other constraints related to the availability of the courses and conflicting timetables that often force suboptimal choices.

This paper intends to introduce an AI-driven Elective Recommendation System that will aid students in making the perfect choice of electives for their curricular needs in alignment with their interests and vision in academics. It uses advanced techniques of machine learning and data analytics to analyze their academic history, interests, and career objectives with the view of offering personalized data-driven recommendations to the students.

A hybrid recommendation approach has been applied, which combines the collaborative filtering method with the content-based filtering technique for the improvement of accuracy. In the process of the collaborative filter, electives are made according to the patterns that are found in other students' similar selections. In the content-based filter, it assesses attributes of courses about topics, prerequisites, and difficulty levels.

The proposed system benefits both students and academic institutions. For students, it simplifies the decision-making process, enhances academic engagement, and supports effective career planning by aligning course selections with individual goals. For institutions, it optimizes course enrollments, improves resource allocation, and reduces dissatisfaction related to elective allocation.

By integrating advanced algorithms with institutional systems, the Elective Recommendation System (ERS) streamlines the elective selection process for students and academic administrators

II. Related and Existing work

Elective Recommendation Systems (ERS) have been proposed as a method of solving problems faced by students in electives. Several techniques have been pursued for the development of ERS, and most of them focus on data-driven methods such as content-based filtering and hybrid approaches where various combinations of methods are used to analyze behavior characteristics and make recommendations to students.

Despite the advancements in ERS, several challenges persist, including data sparsity, cold start problems, and aligning recommendations with career goals. Additionally, while ERS has proven effective in improving academic performance, integrating career aspirations remains a complex task.

The table below summarizes key works related to ERS and the techniques they explored

Technique Reference and Key Findings

Table:1 Machine Learning Techniques

Technique	**Reference**	**Key Findings**
Content-Based Filtering	Gonçalves et al. (2017)	Effective for recommending electives by analyzing course attributes (topics, prerequisites, difficulty). Suitable for systems with limited
Hybrid Methods	Rashid et al. (2018)	Combines content-based filtering with other methods to improve accuracy. Addresses cold start problems by integrating user and course data.
Data Sparsity Issue	Kim et al. (2019)	Highlighted the challenge of data sparsity, which impacts recommendation accuracy, especially for new courses and users.
Career Alignment	Saxena et al. (2020)	Suggested that incorporating career goals can enhance recommendation relevance but increases system complexity.
Improved Academic Performance	Zhao et al. (2021)	Found that students using ERS performed better academically by choosing electives aligned with their interests and strengths.

III. METHODOLOGY

Elective Recommendation Systems (ERS) are designed to assist students in selecting courses that align with their academic goals and interests. These systems leverage various technologies and methodologies to provide personalized recommendations effectively.

To manage and store data, **SQL and MySQL integration** plays a crucial role. MySQL acts as the database management system where data such as student details, course offerings, and elective preferences are stored. SQL queries enable efficient data retrieval and manipulation, allowing the system to fetch, insert, update, or delete records based on user interactions. While this approach ensures data reliability and supports complex queries, challenges like maintaining referential integrity and database normalization must be addressed.

Relational database concepts such as primary keys, foreign keys, and auto-increment features ensure data consistency and minimize redundancy. These concepts create meaningful relationships between tables, linking entities like students, courses, and departments. However, managing complex relationships and maintaining consistent data across tables can be challenging.

The **web interface** is developed using HTML5, CSS, and Bootstrap, providing the structure, design, and responsiveness necessary for an intuitive user experience. Jinja2, a templating engine, enables dynamic content generation, such as looping through student data and displaying recommendations. JavaScript further enhances the interface by enabling interactive features like dynamic table switching and search functionality. While these tools offer a visually appealing and responsive user interface, balancing performance for large datasets remains a challenge.

For backend development, the **Flask framework** is used to handle web routes, manage requests, and render templates. Flask provides flexibility and ease of integration with tools like MySQL, enabling the development of a dynamic web application. The **flask_mysqldb** extension facilitates seamless interaction between the application and the database, allowing for efficient data management. Proper handling of database connections and query optimization is essential to maintain performance, particularly for large datasets.

Machine learning enhances the recommendation process by analyzing patterns in student preferences and course attributes. Cosine similarity, implemented using the sklearn.metrics.pairwise library, calculates the similarity between courses and student profiles, enabling personalized recommendations. Additionally, OpenAI's embedding models generate semantic representations of course descriptions and student preferences, providing

more accurate and meaningful suggestions. However, the effectiveness of these models depends heavily on data quality, and computational intensity can pose challenges.

Flask session management ensures a personalized and seamless user experience by storing temporary data, such as login status and selected electives, throughout a user's interaction with the system. Ensuring the security and privacy of session data is critical, especially when handling sensitive information. Error logging and debugging using Python's logging library also help in identifying and resolving issues, thus making the system more maintainable.

By integrating these technologies and methodologies, the Elective Recommendation System delivers a robust solution to address student needs and institutional challenges. The combination of reliable databases, interactive web design, dynamic backend functionality, and intelligent recommendations ensures an efficient and user-friendly system.

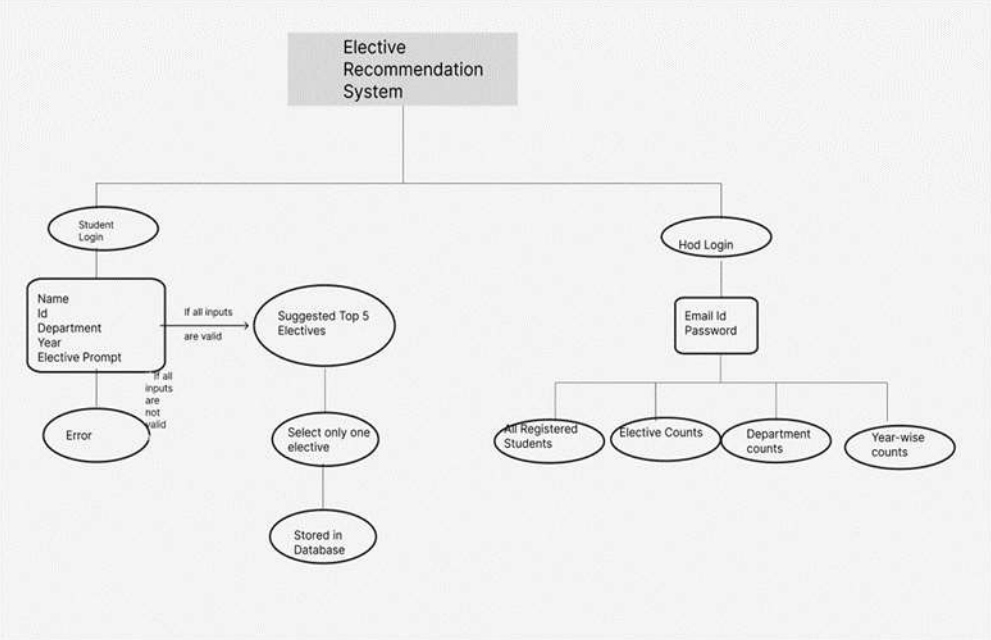

Fig 1: Elective recommendation system

IV. EXCEPTED OUTCOME

The Elective Recommendation System will ease the process of electives selection among students and staff by solving issues such as unclear, random assignment, and unsatisfactory choices. It uses machine learning algorithms, relational databases, and dynamic web frameworks to make personalized recommendations for the student according to his preferences and academic profile. In this way, the students are assured to select their electives appropriately and be satisfied with them, leading to higher satisfaction and better academic performance.

For faculties and administrators, it reduces the complications associated with elective management because the system informs students about preference distribution of students per department per year in enrolling students in elective classes. With these concepts and its elements - the relational database: primary key, foreign key, normalization of the data assures organized consistency as well as quick retrieval. Automated features like elective count tracking and student assignment further reduce the manual effort required to manage electives.

The system is built using HTML5, CSS, and Bootstrap, so it is automatically compatible with everyone's favorite devices and easy to use. Dynamic content rendering through Jinja2 and JavaScript-based enhancements, including table switching and search functionality, help improve the overall user experience of the system, making it visually appealing while also functionally robust. Additionally, the backend, powered by Flask and MySQL, facilitates secure data handling, efficient routing, and seamless integration with machine learning models for generating recommendations.

One of the key outcomes of the system is its ability to provide fair and data-driven elective allocation, addressing common grievances such as unequal distribution or students being separated from their peer groups. By utilizing machine learning techniques like cosine similarity and embedding generation, the system ensures accurate recommendations tailored to individual students. These recommendations also help faculty, particularly Heads of Departments, in making informed decisions about elective assignments, ensuring equitable distribution while meeting departmental requirements.

Moreover, the system promotes scalability and future adaptability. With a modular architecture, new features, such as predictive modeling, real-time feedback, or even expanded datasets, can be incorporated without disturbing existing functionality. In this way, it is sustainable enough to manage electives in highly dynamic academic environments. Overall, the Elective Recommendation System will enhance transparency, efficiency, and satisfaction for all parties involved in the elective selection process.

V. CHALLENGES

Some of the weaknesses perceived in the framework such as, user login, database management, flow design, data sharing, user interface, growth, and confidentiality, are some of the areas which need to be considered in the workings of the system. In doing so, it would be a matter of enhancing security, increasing the efficiency of the database and improving the design of the system so that users can use it easily.

1.Despite their promise, ERS face significant challenges. One prominent issue is data sparsity, where a lack of comprehensive data reduces prediction accuracy. Kim et al. (2019) emphasize the need for robust datasets to enable precise recommendations, posing limitations for new users or electives.

2. Another challenge lies in career alignment. While many ERS prioritize academic performance, Saxena et al. (2020) suggest incorporating career outcomes into recommendations to increase their relevance. However, this integration introduces additional complexity to system design.

VI. CONCLUSION

The Elective Recommendation System will streamline elective selection for the students and faculties, which sometimes lacks clarity, makes arbitrary assignments, and causes unhappiness over election choices. In this system, advanced technologies of machine learning, relational databases, and dynamic web frameworks will make recommendations to a student based on his or her preferences and profile. It ensures that students make wise decisions on electives, which leads to higher student satisfaction and academic outcomes.

For faculty and administrators, the system simplifies the process of managing electives by offering insights into student preferences, department-wise elective distribution, and year-wise enrollment patterns. The integration of relational database concepts, such as primary keys, foreign keys, and normalization, ensures that the data is organized, consistent, and easily accessible. Automated features like elective count tracking and student assignment further reduce the manual effort required to manage electives.

The system's intuitive user interface, built using HTML5, CSS, and Bootstrap, ensures accessibility and ease of use across devices. Dynamic content rendering via Jinja2 and JavaScript-based enhancements like table switching and search functionality improve the overall user experience, making the system both visually appealing and functionally robust. Additionally, the backend, powered by Flask and MySQL, facilitates secure data handling, efficient routing, and seamless integration with machine learning models for generating recommendations.

One of the key outcomes of the system is its ability to provide fair and data-driven elective allocation, addressing common grievances such as unequal distribution or students being separated from their peer groups. By utilizing machine learning techniques like cosine similarity and embedding generation, the system ensures accurate recommendations tailored to individual students. These recommendations also help faculty, particularly Heads of Departments, in making informed decisions about elective assignments, ensuring equitable distribution while meeting departmental requirements.

Moreover, the system promotes scalability and future adaptability. The modular architecture allows for the addition of new features, such as predictive modeling, real-time feedback, or expanded datasets, without disrupting the existing functionality. This makes it a sustainable solution for managing electives in dynamic academic environments. Overall, the Elective Recommendation System will enhance transparency, efficiency, and satisfaction for all stakeholders involved in the elective selection process.

VIII. REFERENCES

1. Ricci, F., Rokach, L., & Shapira, B. (2015). *Recommender Systems Handbook*. Springer. This book provides comprehensive coverage of recommender systems, including algorithms and methodologies for building recommendation models, which is highly relevant for the ERS.
2. Salton, G., & McGill, M. J. (1983). *Introduction to Modern Information Retrieval*. McGraw-Hill. This reference explains the mathematical concepts and applications of cosine similarity, commonly used in recommendation systems.
3. Grinberg, M. (2018). *Flask Web Development: Developing Web Applications with Python*. O'Reilly Media. This book covers Flask, which was used for building the web application in the ERS project. It details how to integrate databases, handle requests, and develop a user-friendly UI.
4. Kline,K.,&Pomeroy,K.(2006). *MySQLTutorial*.O'Reilly Media.A tutorial for MySQL, explaining how to manage databases and implement relational concepts like primary and foreign keys, which are essential for the ERS project.
5. Walton, M. (2016). *Bootstrap: Responsive Web Development*. Packt Publishing. This resource offers insights into using Bootstrap for creating responsive, mobile-friendly websites, which was applied in designing the ERS UI.
6. Friedl, J. E. F. (2006). *Mastering Regular Expressions*. O'Reilly Media. This book provides detailed explanations on how to use regular expressions, an essential tool for input validation in the ERS project.
7. OpenAI. (2021). *OpenAI API Documentation*. Retrieved from https://beta.openai.com/docs/ This official OpenAI documentation provides a guide on how to use embedding models for text similarity, which was a core part of the recommendation algorithm in the ERS.