A Collaborative Filtering Recommendation System for Elective Courses Using Cosine Similarity

Penukulapati Durga Maruthi Vara Prasad[1], Allen Saji[2], Faizan Niyazuddin[3], Syed Yusuf Husian[4], Naymaan Khan[5], Swati Sharma[6]

[1,2,3,4,5] *U.G. Student, Department of Computer Science Engineering, Presidency University School of Engineering, Ittagallpura, Karnataka, India*

[6]*Professor, Department of Computer Science Engineering, Presidency University School of Engineering, Ittagallpura, Karnataka, India*

***Abstract*—We have tackle the problem of elective recommendation at Presidency University, where students often struggle to secure their preferred subjects. Additionally, when they do receive their electives, they frequently find themselves separated from their friends, leading to confusion. Many students are arbitrarily assigned to different electives by their Heads of Department (HoD) or the timetable committee in an attempt to achieve a balanced distribution among various discipline electives. This project employs a Cosine similarity collaborative filtering recommendation system algorithm designed to assist students in selecting their discipline and open electives based on their academic performance. Furthermore, we provide an interactive dashboard for students and an administrative dashboard for HoD's to oversee the recommendations without resorting to arbitrary assignments.**

***Keywords*—Cosine Similarity, Collaborative Filtering, Recommendation System, Streamlit Dashboard**

I. INTRODUCTION

Machine learning recommendation systems have gained significant traction across various applications, from recommending movies and TV shows to suggesting items in online shopping. These systems function by analyzing user preferences and consumption patterns to compute the most suitable items for recommendation [1], [2].

Generally, there are two primary categories of recommendation systems: collaborative filtering and content-based filtering. Collaborative filtering relies on user-item interactions to identify patterns and make recommendations based on the preferences of similar users [3]. This approach can be further divided into two subtypes: user-based collaborative filtering, which recommends items based on preferences from similar users, and item-based collaborative filtering, which suggests items similar to those a user has liked in the past [1]. Userbased collaborative filtering identifies users with similar tastes and recommends items they have liked, while item-based collaborative filtering computes similarities between items based on user ratings [4].

The fundamental equation for collaborative filtering can be expressed as:

$$R_{u,i} = \frac{\sum_{j\in N(i)} R_{u,j}}{|N(i)|} \quad (1)$$

where $R_{u,i}$ is the predicted rating for user u on item i, $N(i)$ represents the set of users who have rated item i, and $|N(i)|$ is the number of such users [5].

On the other hand, item-based filtering computes similarities between items based on user ratings and recommends items that are similar to those already rated by the user. The equation for item-based filtering can be represented as:

$$\text{sim}(i,j) = \frac{\sum_{u\in U} R_{u,i}R_{u,j}}{\sqrt{\sum_{u\in U} R_{u,i}^2}\sqrt{\sum_{u\in U} R_{u,j}^2}} \quad (2)$$

where $\text{sim}(i,j)$ is the similarity between items i and j, and U is the set of users who have rated both items [4].

The students at Presidency University face challenges when instructed to enroll in elective courses while transitioning into higher semesters to fulfill their undergraduate program course credits. The enrollment process has traditionally been based on a generic Microsoft Form circulated among students, which fails to provide individual preferences or information regarding seat availability in chosen courses. Given our dataset, we opted for the cosine similarity algorithm due to its effectiveness in measuring similarity between two non-zero vectors. This method is particularly suitable for our context as it allows us to recommend courses based on students' academic performances.

Our project employs a cosine similarity collaborative filtering recommendation system designed to assist students in selecting their discipline and open electives based on their academic performance. By utilizing historical student enrollment data and academic performance metrics (marks/grades), we aim to provide personalized elective recommendations that enhance student satisfaction and engagement. The architecture comprises separate modules for handling student interactions, HoD management, and faculty management. We utilize SQLite3 for database management and Streamlit for creating interactive front-end dashboards, with algorithmic implementations by the Scikit-learn library.

The cosine similarity algorithm first verifies whether the provided roll number (hashed string) is correct before proceeding to compute recommendations using a cosine collaborative filtering algorithm based on current academic grades. Each student receives a list of ten recommended courses along with real-time seat availability information. The database (SQLite3) maintains enrollment details and updates seat allocations accordingly. The HoD interface provides administrative privileges such as deleting specific student enrollments from a specific course, updating course seat allocations, and downloading enrollment data in CSV file format. Any changes made by HoDs are instantly reflected in both the database and student interface.

II. LITERATURE REVIEW

As the world moves toward an interdisciplinary approach, the need for elective courses has increased exponentially, prompting universities to adapt a diverse curriculum that encompasses not only core courses but also relevant subjects based on students' chosen fields. Elective courses provide diverse intellectual stimulation and cultivate multifaceted curiosity, leading to the development of sophisticated contemporary projects that open global and research opportunities.

Research by Mwelwa and Sooltan Sohawon [6] emphasizes that students' choices in elective courses are influenced by personal interests, career aspirations, and peer recommendations. They found that when students pursue their desired courses, overall satisfaction with learning enhances drastically. This aligns well with our project's goal of providing personalized elective recommendations based on individual academic marks and grades.

Collaborative filtering has emerged as an important technique for generating user-based recommendations across various domains, including movies, shopping, and education. A study by Koren et al. [5] illustrates how collaborative filtering can effectively predict user preferences based on the preferences of similar users. In our project, we have implemented cosine similarity to compare student academic profiles with those of other students who have successfully completed elective courses, thereby enhancing the overall performance of elective course recommendations.

As data becomes increasingly viable, data-driven decisionmaking is recognized as a key factor in expanding higher education to improve student outcomes and overall institutional effectiveness. A report by the National Center for Education Statistics (NCES) highlights the importance of utilizing historical student enrollment data and academic performance metrics (marks/grades) to guide the assignment of elective courses [7].

Despite the functionality of recommendation models, several challenges exist during their execution. Research by Santini et al. [8] has identified common issues such as limited seat allocations, misalignment between students' preferred courses and those assigned to them, and constraints imposed by administrative officials. In our system, we have implemented dynamic database updates for overall seat allocation for each specific course, which is managed by Heads of Departments (HoDs) to resolve any seat availability constraints.

Custom-tailored learning has demonstrated significant enhancements in how students engage with academics, resulting in higher individual success rates. A study by Walkington [9] found that providing customized educational experiences tailored to individual students' needs leads to improved selfefficacy regarding learning and overall motivation resulting in better learning outcomes. By offering personalized elective course recommendations based on individual academic grades and marks, our recommendation model promises greater engagement among students at Presidency University.

III. METHODOLOGY

A. Dataset Collection and Preparation

The dataset comprises 10,600 records provided by the university. Each student's roll number has been converted into a hashed string for privacy protection. The dataset includes course codes, marks scored out of 200, grades (O/A+/A/B+/B/C/D), and grade points (scored out of 10). A total of 24 unique courses are represented:

TABLE I: List of Courses

No.	Course Code
1	CSE228
2	CSE3114
3	CSE3005
4	CSE3115
5	CSE3014
6	CSE3112
7	CSE243
8	CSE3111
9	CSE6002
10	BCA217
11	CSA3020
12	CSE3113
13	CSE3134
14	CSE3001
15	CSE3036
16	CSE5006
17	CSE3106
18	CSE2037
19	CSE2039
20	MAT1002
21	CSE3152
22	CSE2060
23	CSE2015
24	CSE3087

B. Algorithm Implementation

Cosine Similarity measures the similarity between two nonzero vectors [4]. It is defined as:

Cosine similarity is a metric used to measure how similar two non-zero vectors are, regardless of their magnitude. It is particularly useful in recommendation systems where we want to determine the similarity between users or items based on their attributes. The cosine similarity between two vectors:

$$\text{Cosine Similarity}(A, B) = \frac{A \cdot B}{\|A\|\|B\|} \quad (3)$$

Here, $A \cdot B$ represents the dot product, and $\|A\|$ and $\|B\|$ are magnitudes of vectors A and B.

In our project, we utilize cosine similarity to recommend elective courses to students based on their academic performance. The dataset comprises student records, including their roll numbers (hashed for privacy), course codes, grades (scored out of 200), and corresponding grade points.

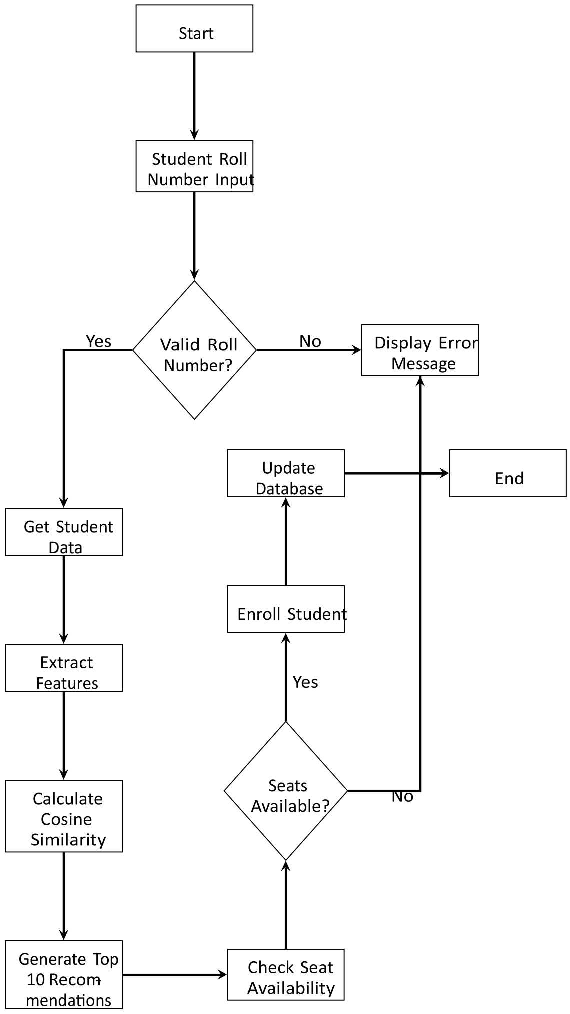

Fig. 1: Flowchart of the Recommendation System

Algorithm 1 Pseudocode for Cosine Similarity Collaborative Filtering Algorithm

```
function GETRECOMMENDATIONS(roll number)
    if not verifyStudent(roll number) then
        return "Invalid Roll Number"
student features ← getStudentData(roll number)
all courses ← getAllCourses()
similarities ← calculateCosineSimilarity(student features, all courses)
recommended courses← getTopCourses(similarities)
    return recommended courses
```

C. Algorithm Structure

The algorithm is structured into the following steps:

- Data Preparation: The dataset is preprocessed to ensure that all numerical values (marks and grade points) are in a suitable format for analysis.

- Verification of Student Input: The algorithm first verifies whether the provided roll number corresponds to a valid entry in the dataset.
- Data Representation: We construct a user-item matrix where each row corresponds to a student and each column corresponds to a course.
- Cosine Similarity Calculation: For every student's feature vector, we compute its cosine similarity with each course vector using:

cosine similarity(*student vector, course vector*)

- Recommendation Generation: After calculating similarities, we rank courses based on their similarity scores and select the top 10 recommendations for each student.
- Dynamic Updates: The system dynamically updates enrollment data in the SQLite database as students enroll in courses based on recommendations.

D. Database Implementation

We utilized SQLite3 for database management containing all 24 unique course codes with an initial seat allocation of 300 per course. The database records student roll numbers (hashed strings) along with opted courses.

- student.py: Collects student enrollment data.
- hod.py: Displays existing records in the database; allows modifications such as deletions of specific enrollments, updation of seat allocation, download option available.
- faculty.py: Enables faculty members to view enrollments for specific courses with download options available.

IV. USER INTERFACE

Developed using Streamlit, the interface provides three main dashboards for different user roles:

A. Student Interface

The student interface presents ten course recommendations based on academic grades, allowing students to view and select their preferred courses. Key features include:

- Real-time course recommendations
- Available seat information
- Simple enrollment process

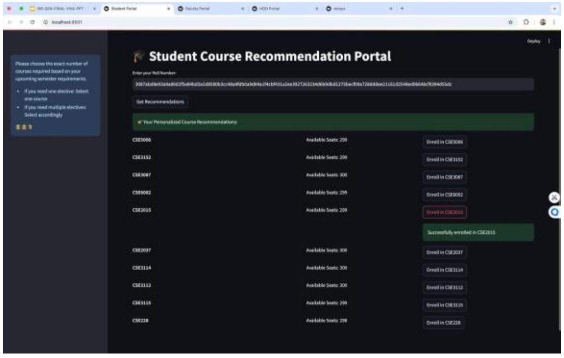

Fig. 2: Student Dashboard showing course recommendations and enrollment options

B. HoD Interface

The HoD interface provides comprehensive administrative controls for managing course enrollments. Key features include:

- Enrollment monitoring
- Student record management
- Seat allocation updates
- CSV export functionality

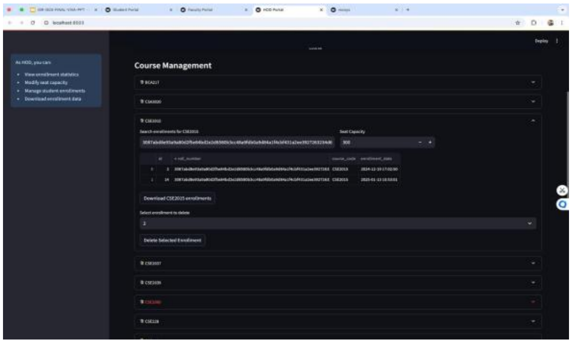

Fig. 3: HoD Dashboard with administrative controls and enrollment management

C. Faculty Interface

The faculty interface enables course-specific enrollment monitoring with data export capabilities. Key features include:

- Course-specific enrollment views
- Download options for records
- Real-time updates

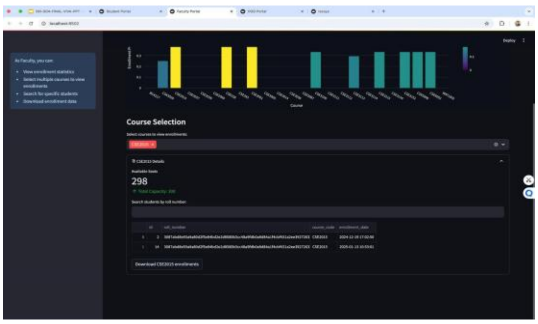

Fig. 4: Faculty Dashboard showing course-specific enrollment details

V. RESULT

The academic course recommendation system developed for Presidency University effectively utilizes machine learning and database management to enhance elective allocation for students. By employing a collaborative filtering approach and cosine similarity, the system personalizes recommendations based on historical performance data from 10,600 student records across 24 courses. Feedback indicates increased student satisfaction with elective choices, addressing previous issues with course assignments. The integration of SQLite3 allows for efficient real-time seat allocation, facilitating concurrent enrollments without delays. Role-based access control improves user experience by tailoring interfaces for students, faculty, and HODs, while real-time visualization tools like Plotly and Streamlit provide insightful analytics for decisionmaking. The automated seat allocation system ensures equitable access to popular courses, contributing to the broader discourse on educational technology by demonstrating the practical application of data-driven decision-making in academic administration. The modular architecture of the project enhances scalability and maintainability, ensuring long-term reliability and efficiency.

VI. CONCLUSION

By successfully implementing cosine similarity, along with an interactive dashboard with the help of Streamlit, students can now enroll in their desired elective course based on their academic grades plus the number of available seats allocated for that particular course along with their friends. A separate dashboard for HoD enables them to monitor all the enrollments to prevent any form of conflict which arises during enrollment. All modules (student.py, database.py, recsys.py, hod.py) have been interconnected to ensure real-time dynamic interactivity.

REFERENCES

[1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, 2005.

[2] F. Ricci, L. Rokach, and B. Shapira, *Recommender Systems Handbook*. Springer, 2011.

[3] S. Xiaoyuan and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in Artificial Intelligence*, vol. 2009, p. Article ID 421425, 2009, [Online]. Available: https://doi.org/10.1155/2009/421425.

[4] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the 10th International Conference on World Wide Web*. ACM, 2001, pp. 285–295.

[5] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer Science Review*, vol. 29, no. 1, pp. 30–80, 2009.

[6] M. Kapambwe and M. S. Sohawon, "Educational administration and management: Issues and perspectives," *International Journal of Scientific Research in Management, Science and Technology*, vol. 3, no. 4, p. Article ID 199, 2020.

[7] National Center for Education Statistics (NCES), "Data-driven decision making in higher education," 2019.

[8] S. Santini *et al.*, "Comparing institutional, teaching, and student factors in relation to satisfaction," *Educational Administration Quarterly*, vol. 53, no. 2, p. Article ID 00986283241265741, 2017.

[9] C. Walkington, "Personalizing learning: The role of student engagement," *Educational Psychologist*, vol. 48, no. 3, pp. 165–178, 2013.