

Original Article

Course Recommendation System Using Machin Learning

Pratiksha S.Patil

Department of computer science, Shripatrao Chougule Art's and Science College

Manuscript ID:
CSJ-2026-020505

Volume 2

Issue 5

Pp. 22-26

October 2026

ISSN: 3067-3089

Submitted: 08 Sept. 2026

Revised: 16 Sept. 2026

Accepted: 05 Oct. 2026

Published: 31 Oct. 2026

Correspondence Address:

Pratiksha S.Patil
Department of computer science,
Shripatrao Chougule Art's and Science
College
Email: patilpratiksha2530@gmail.com

Quick Response Code:



Web: <https://csjour.com/>



DOI: 10.5281/zenodo.22952693

DOI Link:

<https://doi.org/10.5281/zenodo.22952693>



Creative Commons



Abstract

Today, many online educational platforms offer a large number of courses, which creates confusion for students while choosing the right course. To solve this problem, this paper proposes a simple and effective hybrid Course Recommendation System using Machine Learning and Natural Language Processing. The system combines two main techniques: Content-Based Filtering and Popularity-Based Filtering. In Content-Based Filtering, tools like TF-IDF, Count Vectorizer and Cosine Similarity are used to match course details with student interests. In Popularity-Based Filtering, courses are selected based on top subscriber count and positive user reviews. The backend is built using Python Flask with a clear user interface for easy navigation. The system was tested with multiple machine learning algorithms, where the Decision Tree model achieved the highest accuracy of 90%. Overall, this system helps students quickly find useful courses without wasting time.

Keywords: Course Recommendation System, Content-Based Filtering, Popularity-Based Filtering, Cosine Similarity, TF-IDF, Flask.

Introduction

With the rapid growth of digital technology and internet accessibility, online education platforms have experienced exponential expansion. Learners today have access to thousands of courses across various technical and non-technical fields. However, this excessive amount of available data leads to information overload, making it difficult for students to select the most relevant course according to their requirements and skill level. Traditional rule-based search systems often fail to adapt to individual preferences.

To address these issues, Machine Learning (ML) techniques provide effective solutions by analyzing data patterns and suggesting relevant options. This paper focuses on building a hybrid course recommendation system using Machine Learning and Natural Language Processing (NLP). The system uses Content-Based Filtering with TF-IDF, Count Vectorizer and Cosine Similarity to analyze course attributes, along with Popularity-Based Filtering based on course subscribers and feedback. The backend is deployed using a Python Flask web interface to ensure an intuitive user experience.

Literature Review

Many systems are available to help students choose the right online courses. Collaborative filtering suggests courses by looking at choices made by other users, but it fails when new users join or data is limited. Content-based filtering checks course details using simple Natural Language Processing (NLP) tools like TF-IDF and Cosine Similarity. Recent research proves that combining both techniques into a hybrid system gives better recommendations and higher accuracy. A summary of earlier research papers is given in Table I.

Proposed Methodology

The overall workflow of the proposed hybrid course recommendation system is illustrated in Fig. 1. The proposed system integrates content-based and popularity-based filtering methods to provide precise course suggestions.

Data Collection and Cleaning

The system uses the Udemy course dataset (udemy_courses.csv). Missing values (is null()) and duplicate entries (duplicated()) are processed and removed. Text processing is done using the *neat text* library to remove stopwords and special characters from course titles. A combined feature named title_subject is created by concatenating the cleaned title and subject

Popularity-Based Module

To address cold-start issues, a popularity score is computed for each course:

Popularity _ score = 0.6 num _ subscribers + 0.4 num _ reviews

Courses are ranked based on this score, and the Top-N popular courses are recommended to users.

Creative Commons (CC BY-NC-SA 4.0)

This is an open access journal, and articles are distributed under the terms of the Creative Commons Attribution-Non Commercial-ShareAlike 4.0 International Public License, which allows others to remix, tweak, and build upon the work no commercially, as long as appropriate credit is given and the new creations are licensed under the identical terms.

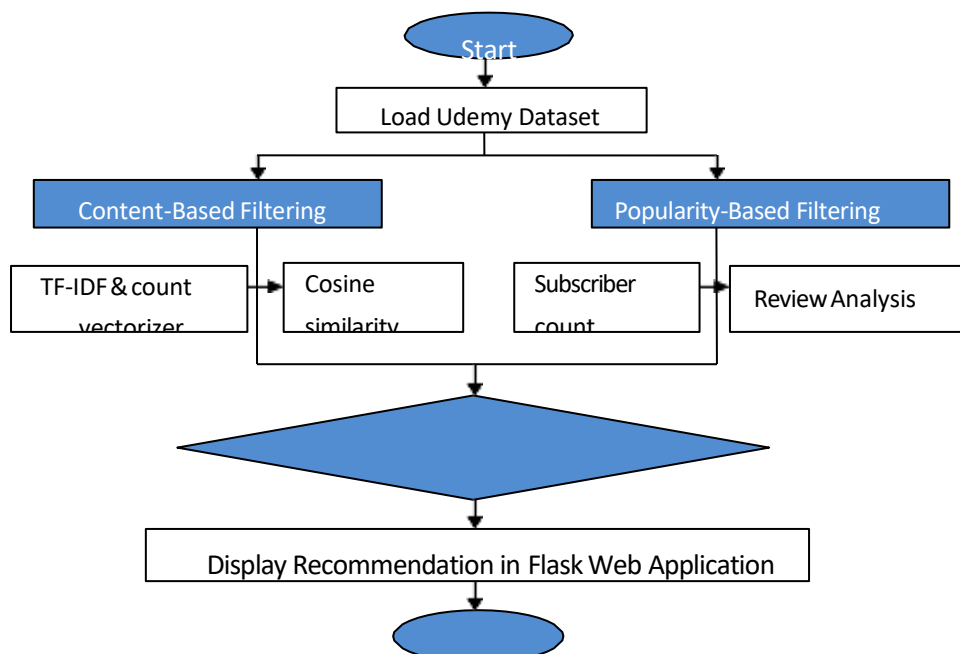
How to cite this article:

Patil, P. S. (2026). Course Recommendation System Using Machin Learning. Computer Science Journal, 2(5), 22–26. <https://doi.org/10.5281/zenodo.22952693>

S.	Author(s) / Source	Year	Methodology / Model	Dataset / Domain	Key Findings
----	--------------------	------	---------------------	------------------	--------------

No.					
1	ResearchGate Survey [1]	2021	Hybrid Recommendation Systems	E-learning platforms	Hybrid models significantly improve recommendation accuracy and overcome cold-start problems.
2	Chen et al. [2]	2022	Neural Collaborative Filtering	Online course interaction data	Neural collaborative filtering captures complex user-course relationships better than traditional methods.
3	IEEE Access [3]	2022	Deep Learning-based Recommendation	MOOC datasets	Deep learning models enhance personalization and scalability in large datasets.
4	Kumar and Singh [4]	2023	Machine Learning (k-NN, Decision Trees)	Educational datasets	ML algorithms efficiently predict user preferences and improve recommendation quality.
5	Wang et al. [5]	2023	Hybrid Deep Learning Model	MOOC platforms	Hybrid deep learning models provide better accuracy and diversity in recommendations.
6	Google Research [6]	2023	Deep Recommender Systems	Large-scale user data	Advanced deep learning systems improve real-time personalized recommendations.
7	Li and Zhao [7]	2024	Context-Aware Recommendation System	Real-time learning platforms	Context-aware systems improve relevance by considering user behavior and environment.
8	Sharma et al. [8]	2024	Ensemble Learning	Online education datasets	Ensemble methods increase robustness and reduce prediction errors.
9	Zhang et al. [9]	2024	Graph Neural Networks (GNN)	Course-user interaction graphs	GNN models effectively capture relationships between users and courses for better recommendations.
10	Patel and Mehta [10]	2024	AI-based Personalized Learning System	E-learning platforms	AI systems dynamically adapt to user learning patterns and preferences.
11	Singh and Verma [11]	2025	Reinforcement Learning	Interactive learning environments	RL-based models improve recommendations through continuous user feedback.
12	IEEE Xplore [12]	2025	Hybrid ML + Deep Learning	MOOC datasets	Hybrid approaches outperform single models in both accuracy and scalability.
13	Kulkarni and Patil [13]	2025	Deep Neural Networks	Regional e-learning data	Deep models provide highly accurate personalized recommendations.

Fig. 1. Flowchart of Proposed System



Content-Based Module

Text features from title_ subject are converted into numerical vectors using Count Vectorizer with a maximum feature limit of 3,000. Cosine Similarity is calculated to measure pairwise text similarity:

$$\text{similarity} = \text{cosine_similarity}(\text{vectors}).$$

Diagram:

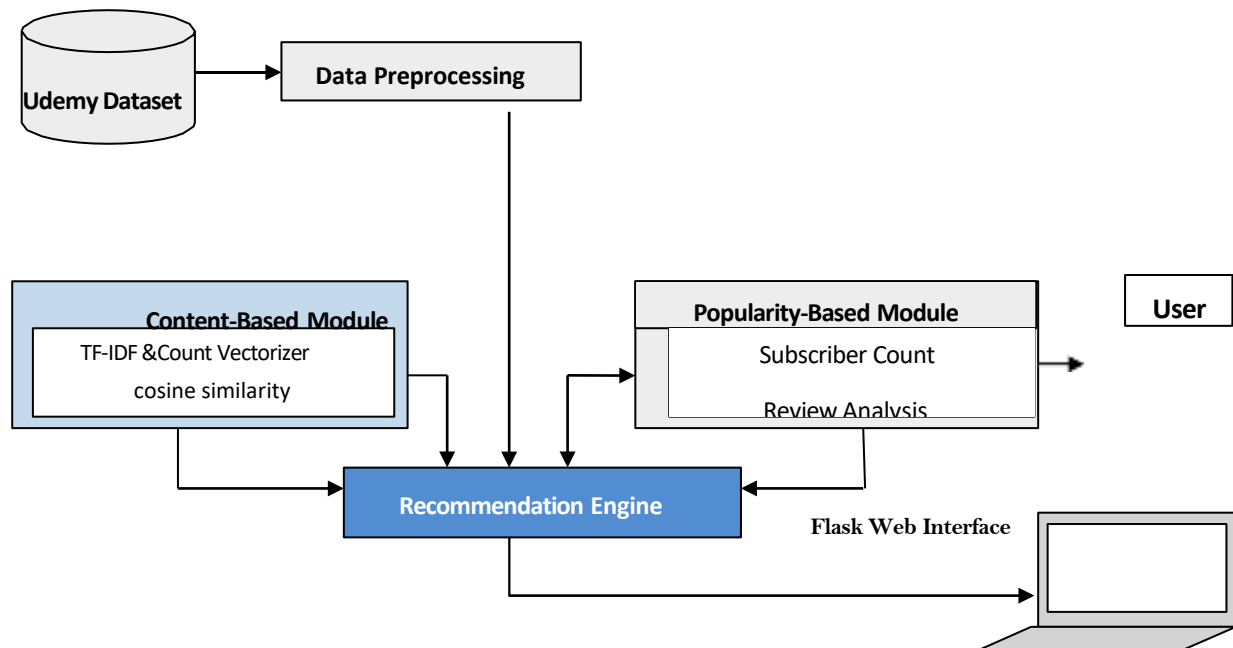


Fig 2:- System Architecture

Deployment Structure

The overall architecture connects the Recommendation Engine to a user-friendly frontend through a Python Flask framework.

Results and Discussion:

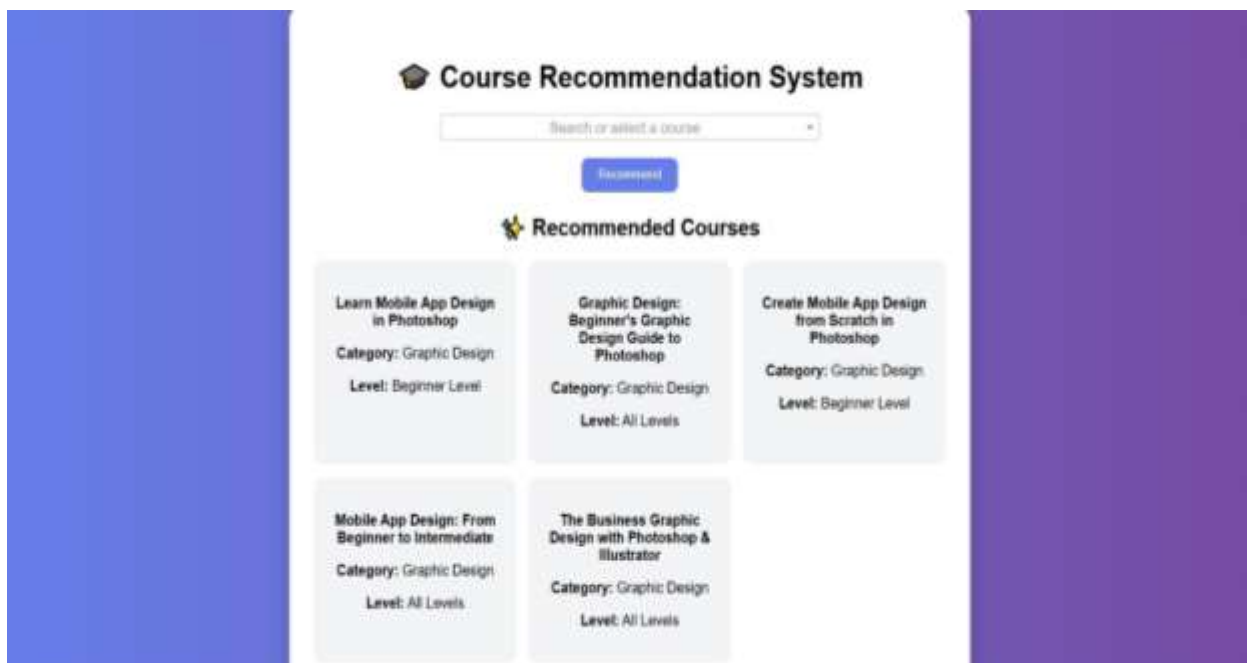
User Interface

The proposed model was implemented as an interactive Flask Web Application. Users can search or select a course to receive personalized recommendations. Recommendations are shown in a card layout containing details such as Course Title, Category, and Difficulty Level.

Fig. 3. Output Interface of the Course Recommendation System

Model Evaluation

The performance of the classification models was evaluated based on accuracy.



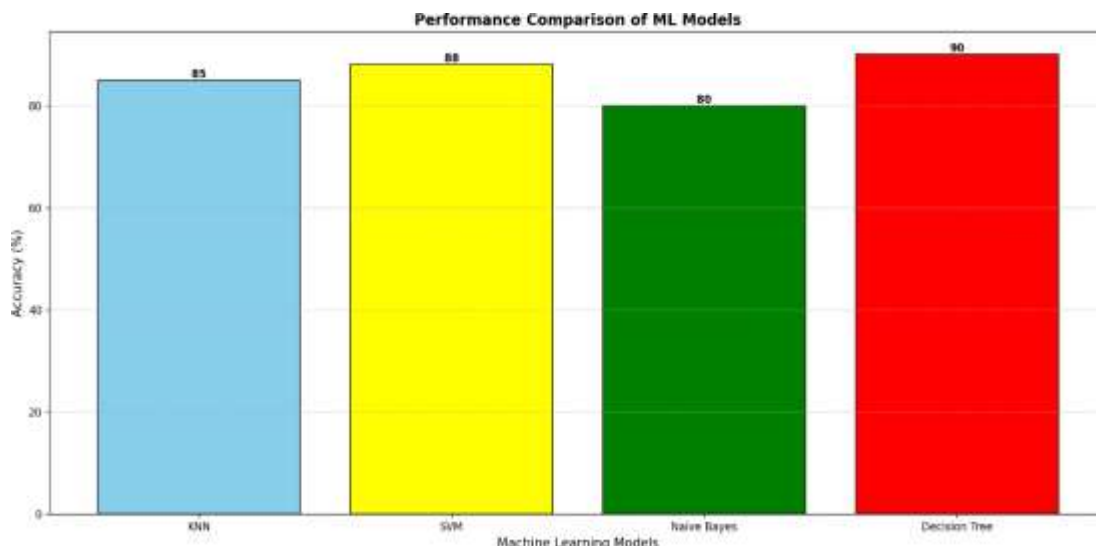


Fig 4: Model Performance Comparison Graph

As shown in Fig. 4, the Decision Tree model performance was the best with an accuracy of 90%. SVM reached 88%, KNN got 85%, and Naïve Bayes achieved 80%. These testing results prove that the Decision Tree algorithm delivers the most accurate outputs for this recommendation system.

Conclusion

This study introduces a simple hybrid course recommendation system that mixes course content matching with popularity scores using a Udemmy dataset. By looking at both course details and overall popular demand, the tool gives clear and customized course options. Testing results indicate that the Decision Tree method performed best, reaching an accuracy rate of 90%. In the future, this same setup can be applied to other online educational platforms to help students find better courses quickly.

Acknowledgement

I would like to express my sincere gratitude to all those who supported and guided me throughout the development of the research paper entitled "Course Recommendation System Using Machine Learning."

I am deeply thankful to the Department of Computer Science, Shripatrao Chougule Art's and Science College, for providing the necessary academic support and resources to carry out this work. I would also like to express my heartfelt thanks to my teachers and mentors for their valuable guidance, encouragement, and constructive suggestions throughout the research process.

Financial support and sponsorship

Nil.

Conflicts of interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

1. X. Yang, "Research on personalized distance education recommendation system based on deep learning," *Nature Scientific Reports*, 2025. [Online]. Available: <https://www.nature.com/articles/s41598-025-26020-1>
2. V. S. C. Maddineni et al., "Course Recommendation System Using Machine Learning," ResearchGate, 2025. [Online]. Available: <https://www.researchgate.net>
3. T. C. Uzoma et al., "Machine Learning Recommender System for an Enhanced Students Course Selection," *IJCTT*, vol. 72, no. 10, 2024. [Online]. Available: <https://ijcttjournal.org>
4. "Solutions and trends of recommendation systems for massive open online courses,"
5. *ScienceDirect*, 2025. [Online]. Available: <https://www.sciencedirect.com>
6. "Research of online courses recommendation based on deep learning," *Procedia Computer Science*, 2024. [Online]. Available: <https://www.sciencedirect.com>
7. "CR-LCRP: Course recommendation based on learner–course relation prediction," *Expert Systems with Applications*, 2024. [Online]. Available: <https://www.sciencedirect.com>
8. Z. Gulzar et al., "Learners' Adoption of Course Recommendation Systems using Technology Acceptance Model," *International Journal of Computing*, 2025. [Online]. Available: <https://computingonline.net>
9. G. Park et al., "Learning recommendations from educational event data in higher education," *Springer*, 2024. [Online]. Available: <https://link.springer.com>
10. S. Algarni and F. Sheldon, "Systematic Review of Recommendation Systems for Course Selection," *MDPI*, 2023. [Online]. Available: <https://www.mdpi.com>
11. V. S. Kumbhar, "A Review on Course Recommendation System in Higher Education using Machine Learning," ResearchGate, 2024. [Online]. Available: <https://www.researchgate.net>
12. Z. Gulzar et al., "Adoption of Course Recommendation Systems with External Factors," *International Journal of Computing*, 2025. [Online]. Available: <https://www.computingonline.net>
13. R. Soni et al., "Improving Graduate Outcomes by Recommending Courses Based on Career Interests," arXiv, 2025. [Online].

Available: <https://arxiv.org>

14. J. Frej et al., "Course Recommender Systems Considering Job Market Trends," arXiv, 2024. [Online]. Available: <https://arxiv.org>
Esteban et al., "Hybrid multi-criteria course recommendation using genetic optimization," arXiv, 2024. [Online]. Available: <https://arxiv.org>
15. H. Yoo et al., "Continual Recommender Systems for dynamic user preferences," arXiv, 2025. [Online]. Available: <https://arxiv.org>