Machine Learning based MOOC Course Recommendation System

Mohammed Suhail Shaikh[1]; Rishabh Nair[2]; Dr. Mohammed Zafar Shaikh[3]; Dr. Asmatara Shaikh[4]; Ujwala Ravale[5]

[1]Dept. of Computer Engineering SIES Graduate School of Technology Navi Mumbai, India

[2]Dept. of Computer Engineering SIES Graduate School of Technology Navi Mumbai, India

[3]Shree Bhagubhai Mufatlal Polytechnic Mumbai, India

[4]Associate Professor Parle Tilak Institute of Management Mumbai, India

[5]Assistant Professor SIES Graduate School of Technology Navi Mumbai, India

Corresponding Author Email: mzshaikh2003@yahoo.co.in

Abstract— The use of Massive Open Online Courses (MOOCs) by students to advance their knowledge and abilities has grown in popularity in the modern world. Students may find it challenging to select the best course for their needs due to the variety of online learning platforms and courses. In this project, we suggest a mechanism for recommending Mooc courses that will assist students in finding the most appropriate course for a certain subject. According to user reviews, the algorithm will trawl through different online learning sites and choose courses. To help students choose the best courses, the recommendation system will take into account variables including subject, level of difficulty, course length, and course rating. Students can save time and effort in looking for the appropriate information by utilizing this approach. The suggested system may be a useful resource for helping students choose top-notch courses that fit their learning goals.

I. INTRODUCTION

In recent years, the Massive Open Online Course (MOOC) market has boomed, with millions of participants attending courses via multiple platforms. The sheer amount of courses offered on various platforms, on the other hand, might be overwhelming for consumers, making it difficult to choose the correct course to fit their needs. To overcome this issue, our re- search intends to develop an adaptive recommendation system for MOOCs. The system employs collaborative filtering and time series analysis to present users with personalized sugges- tions based on their interests and previous usage patterns. The suggestions are created utilizing a web-based platform built using the Streamlit framework and the Python programming language.

The suggested system allows users to utilize a command line to search for courses and offers a list of courses from various MOOC sites. Users may then choose a course to learn more and receive personalized suggestions based on their usage history and interests. A collaborative filtering technique is used by the system to discover comparable users and propose courses that have been highly rated by those users. Furthermore, time series analysis is used by the system to identify trends in user behavior and make recommendations based on those trends. Overall, the adaptive recommendation system strives to improve MOOC platform user experience by offering personalized and relevant course suggestions, making it simpler for users to discover and attend courses that fit their needs.

The adaptive MOOC recommendation system has the potential to change the way consumers navigate and obtain courses. The technology attempts to improve the user experience and promote engagement with the MOOC platform by providing personalised suggestions based on users' interests and his- torical usage patterns. This, in turn, can lead to increased completion rates and improved platform satisfaction. Further- more, the system may help users find and pick courses that are relevant to their professional goals, enhancing their skill set and employability. As the popularity of MOOCs grows, the development and deployment of adaptive recommendation systems will be vital to the platform's growth and success.

Our solution streamlines the course-finding process, saving students time and effort in looking for a suitable course across numerous platforms. The project was written in Python and required the use of a web crawler to collect course data.

II. LITERATURE SURVEY

Fatiha Bousbahia et. al. outlines a case-based recom- mender system created exclusively for MOOCs[1]. The au- thors describe how their system works and the recommenda- tion algorithm, which is based on case-based reasoning. They also address the system's advantages, such as its capacity to deliver personalised suggestions for students, increase student engagement, and improve the overall learning experience in MOOCs. The study finishes by discussing future work and the possibility of merging this approach with other MOOC platforms. Overall, this research gives useful insights into the development of MOOC recommender systems and demon- strates the potential for improving students' online learning experiences.

Kahina Rabahallah et. al. [2] has proposed a system built on ontology and collaborative filtering approaches based on memory. The authors believe that because of the huge amount of accessible information, MOOCs create difficulty for learners, and recommenders can assist learners in navigating and finding relevant courses. The suggested system in the study employs an ontology to describe domain knowledge and allow reasoning about course material. The authors also propose courses to learners based on their interests and behaviour using a memory-based collaborative filtering technique. The system is tested using real data from a MOOC platform, and the findings reveal that it beats other cutting-edge recommender systems.

Vishal Garg et. al [3] have designed a MOOC recommender system that includes both content-based and collaborative filtering approaches. To provide course suggestions for learners, the system employs machine learning methods such as Naive Bayes, Decision Tree, Random Forest, and Gradient Boosting. The authors also created an evaluation methodology to assess the system's effectiveness. The suggested hybrid recommender system outperformed both the content-based and collaborative filtering techniques [3] independently, according to the results. According to the findings of the study, the suggested method delivers a more accurate and personalized learning experience for MOOC participants, resulting in higher engagement and completion rates.

Rodrigo Campos et. al. [4] has designed a web-based recom- mendation system for knowledge reuse in MOOC ecosystems proposed in this study. A user modeling component, a learning object modeling component, and a recommendation engine are all part of the architecture. The learning object modeling component analyses learning objects based on their metadata, whereas the user modeling component develops a user profile based on the user's learning history. Based on the user's learning history and preferences, the recommendation engine provides personalized suggestions for them. The suggested architecture seeks to increase recommendation quality and knowledge reuse in MOOC ecosystems.

YunChou Li et. al. [5] proposed a fusion recommender system for MOOCs that combines several approaches such as content-based filtering, collaborative filtering, and association rules. It seeks to present learners with personalized suggestions based on their preferences, learning styles, and behavioral patterns. To analyze user data and provide suggestions, the system leverages machine learning methods such as k-NN and association rule mining. In terms of accuracy, coverage, and diversity, the experimental findings suggest that MOOC-FRS outperforms typical recommender systems. The system also has a user-friendly interface that allows learners to engage with it and customize their suggestions. Overall, MOOC-FRS has the potential to improve the efficacy and efficiency of MOOCs by delivering personalized and relevant learning resources to participants.

Jonice Oliveria et. al. [6] paper offers a recommendation system that recognizes learners' knowledge gaps and delivers personalized suggestions based on those gaps. To represent the learner's knowledge state and course material, the system employs a knowledge representation model. The system then uses the knowledge gap detection algorithm to identify the gaps and produce appropriate suggestions. The suggested system outperforms the baseline recommendation algorithms in terms of accuracy, recall, and F1 score, according to the assessment findings. The method can assist learners in filling knowledge gaps and improving learning outcomes in MOOCs.

Yanxia Pang et. al. [7] proposes a mix of collaborative filter- ing and time series analysis, this work presents an adaptive recommendation system for Massive Open Online Courses (MOOCs). To forecast future course suggestions for each user, the system employs a dataset combining user behavior and course data. Time series analysis is used to anticipate the popularity of courses over time, while collaborative filtering is used to offer courses that are similar to those already taken by the user. The system is tested on two separate datasets, and the findings reveal that it outperforms other standard recommendation systems. The authors propose that this hybrid technique be used in other e-learning platforms to give users personalized course recommendations. The study offers useful insights into the use of data-driven initiatives in the education sector.

III. DESIGN AND ARCHITECTURE

A recommendation system is one that attempts to anticipate or filter preferences based on the user's selections. A similar system, implemented using Python, proposes courses based on the user's favorite or searched courses.

The flowchart depicts the interaction of a user with a system that delivers course suggestions. The user starts the process by picking courses of interest. Based on the courses selected, the system evaluates the user input and offers course recommendations. These recommendations are then shown to the user, who can comment on them. This feedback is used by the system to update the user profile and generate new recommendations. The system allows the user to view their course progress and completion status. Finally, the procedure concludes

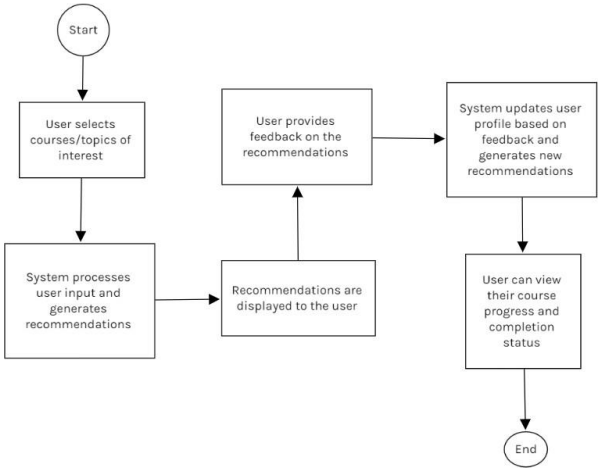

Fig. 1. General Flowchart

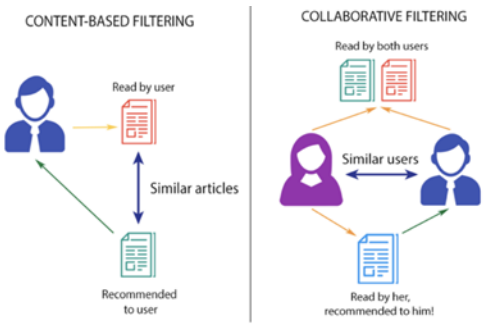

Fig. 2. Recommendation System

CONTENT-BASED FILTERING: In the context of our project, the MOOC course-based recommendation system, we employ content-based filtering methods. These techniques rely on item descriptions and user preference profiles to make personal- ized course recommendations. This approach is particularly effective when we have detailed information about the courses (such as name, location, and description) but limited informa- tion about the users.

Our system utilizes keywords to describe the courses and constructs user profiles to capture their preferences. Essen- tially, we treat the recommendation process as a user-specific classification challenge, training a classifier to discern what types of courses a user is likely to favor based on course features.

One key feature of our approach is that it doesn't require users to sign in to generate their profiles; it adapts to their preferences as they interact with the platform. To make rec- ommendations, we compare various course candidates with those courses that the user has previously rated or is currently exploring. We then recommend the best-matching courses based on this comparison. This methodology finds its origins in the fields of information retrieval and information filtering, and it forms the core of our MOOC course recommendation system.[8]

COLLABORATIVE FILTERING: We leverage collaborative filtering as a key technique for building effective recommender systems. Collaborative filtering can be understood in two ways: a narrower and a more general sense.

In the specific, modern interpretation, collaborative filtering is a method for automatically predicting a user's interests. This prediction is made by gathering preference and taste informa- tion from a collective of users who collaborate in providing data. The fundamental assumption underlying collaborative filtering is that if person A shares similar preferences with person B on one topic, they are more likely to share similar preferences on other topics as well. To illustrate, in the context of our MOOC course recommendation system, collaborative filtering enables us to predict which courses a user might like based on a partial list of their preferences or dislikes related to other courses.[9]

IV. ALGORITHM AND PROCESS DESIGN

Algorithms used are:

• WEB CRAWLER:

Also known as a spider or spider bot, which is often referred to simply as a crawler. This internet bot plays a crucial role in systematically navigating the World Wide Web. Typically, search engines operate these crawlers to accomplish web indexing, a process often called web spidering.

To keep web content up-to-date or to build indices of content from various websites, web search engines, and certain other online platforms rely on web crawling or spidering software. These web crawlers function by replicating web pages for subsequent processing by a search engine. Once the pages are duplicated, the search engine indexes them, making it possible for users to conduct more efficient and accurate web searches. [10]

• COSINE SIMILARITY:

Cosine Similarity emerges as a vital measure for as- sessing the similarity between two vectors within an inner product space, both of which are non-zero. Cosine similarity is derived from the cosine of the angle formed by these vectors, specifically, it is

computed as the dot product of the vectors divided by the product of their magnitudes. It's worth noting that the cosine similarity is independent of the actual vector magnitudes; its focus lies solely on the angle between them. Cosine similarity values always fall within the range of -1 to 1. For instance, when dealing with vectors that are proportional, the cosine similarity yields a value of 1. Conversely, if the vectors are orthogonal, the similarity score is 0, and for vectors that are opposite in direction, the similarity is -1. In some situations where vector com- ponent values cannot be negative, the cosine similarity is restricted to the range of [0,1].

$$S_c(A,B) = \cos\theta = \frac{A.B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} {A_i}^2 \sum_{i=1}^{n} {B_i}^2}} \quad (1)$$

In the context of our MOOC course-based recommenda- tion system project, consider this scenario: in information retrieval and text mining, a technique is employed where each word is assigned a unique coordinate. A document, in turn, is represented as a vector that encapsulates the frequency of each word's occurrence within that docu- ment. When we apply cosine similarity in this context, it provides a valuable metric for gauging the similarity between two documents. This metric allows us to assess how closely related two documents are in terms of their subject matter, all while disregarding the variations in document length.[11]

V. RESULTS

We used streamlit to start a local host where we will run the website.

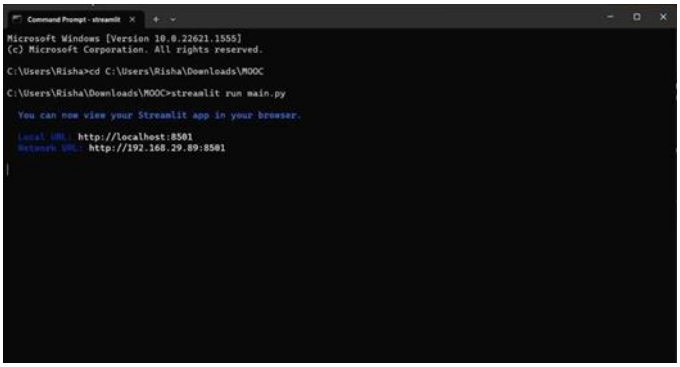

Fig. 3. Local Hosting

This is the main page of the website. Here on the search bar we can search for the courses for the dataset that we scraped. The dataset consists of over 3000 courses from Coursera and Udemy.

After you enter your subject, our website will fetch you results from the datasets After searching the system will show all the recommended courses

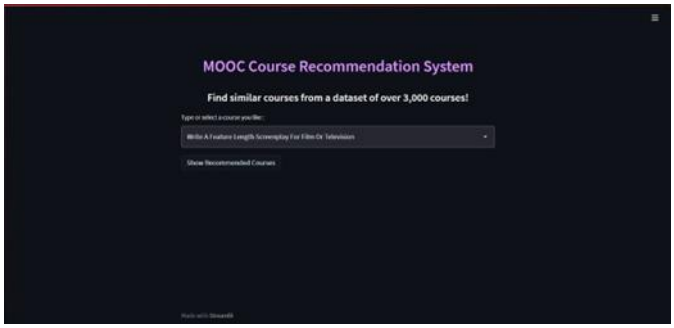

Fig. 4. Main Page

Fig. 5. Searching for C++ courses

Fig. 6. Searching for Python courses

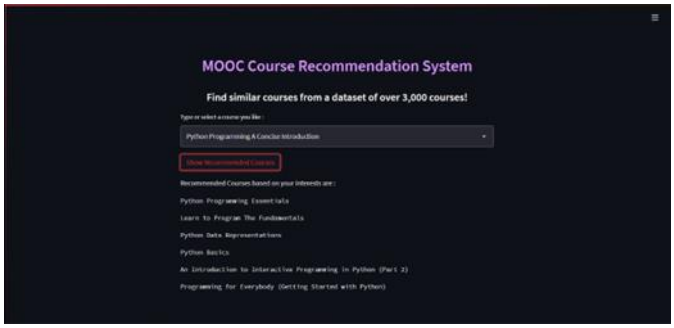

Fig. 7. Recommended Courses

V. CONCLUSION

Finally, the suggested MOOC recommender system has demonstrated promising accuracy and efficiency results. The system can efficiently offer appropriate courses to users while also addressing their individual knowledge gaps by integrating multiple strategies such as collaborative filtering, content- based filtering, and knowledge gap diagnosis. However, there are still certain restrictions, such as the cold-start difficulty for new users and data scarcity in some sectors.

This project presents a novel strategy for MOOC recommen- dation systems to address the limitations of existing systems and cover the research gap. The suggested system incorporates a hybrid paradigm of content-based filtering and collaborative filtering, using the advantages of both techniques. The system also includes an adaptive recommendation algorithm that can adjust to changing user preferences over time. The experimen- tal findings reveal that the proposed system surpasses existing systems in terms of suggestion accuracy and variety.

Future work might concentrate on overcoming these restric- tions by embracing new data sources such as social media and user interactions, as well as experimenting with new ap- proaches such as deep learning and reinforcement learning. To improve the user experience, the system might be developed to offer personalized learning routes and adaptive learning experiences. Overall, the suggested MOOC recommender sys- tem has the potential to dramatically increase online learning accessibility and efficacy.

REFERENCES

1. F. Bousbahia and H. Chorfia, ”MOOC-Rec: A Case Based Recom- mender System for MOOCs,” 2018 3rd International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), pp. 1-5, Sousse, Tunisia, 2018
2. K. Rabahallah, L. Mahdaoui and F. Azouaou, ”Case-Basedmmender System using Ontology and Memory-based Collaborative Filtering,” 2019 3rd International Conference on Computer Science and Engineer- ing (UBMK), pp. 253-258, Antalya, Turkey, 2019.
3. V. Garg and R. Tiwari, ”Hybrid massive open online course (MOOC) recommendation system using machine learning,” 2016 International Conference on Recent Trends in Engineering, Science and Technology
4. - (ICRTEST), pp. 1-6, Hyderabad, 2016.
5. R. Campos, R. P. dos Santos, and J. Oliveira, ”Web-Based Recommenda- tion System Architecture for Knowledge Reuse in MOOCs Ecosystems,” in IEEE International Conference on Information Reuse and Integration (IRI), pp. 218-225 Salt Lake City, UT, USA, 2018.
6. Y. Li and H. Li, ”MOOC-FRS: A new fusion recommender system for MOOCs, ”International Conference on Identification, Authentication and Cybersecurity (ICIDACS), pp. 163-166 Chongqing, China, 2017.
7. R. Campos, R. P. dos Santos and J. Oliveira, ”A Recommendation System based on Knowledge Gap Identification in MOOCs Ecosystems,” IEEE International Conference on Information Reuse and Integration for Data Science (IRI), pp. 231-238, Los Angeles, CA, USA, 2019.
8. Y. Pang, W. Liu, Y. Jin, and H. Peng, ”Collaborative filtering and time series,” Computer Applications in Engineering Education, vol. 26, no. 6, pp. 1855-1862, June. 2018.