EduGuide: A Hybrid NLP-Driven Course Recommendation System for Personalized Online Learning Using TF-IDF and Machine Learning Techniques

Prof. Sujay S[1], Harshitha G[2], Monisha M[3]
[1,2,3] Department of MCA Surana College (Autonomous) Kengeri
doi.org/10.64643/IJIRTV13I3-207439-459

***Abstract*—This study presents the development and evaluation of a Hybrid Course Recommendation System using NLP and Machine Learning techniques to assist learners in identifying relevant online courses from large educational platforms.To fill these gaps,this work proposes a hybrid course recommendation system,which combines the machine learning and Natural Language Processing(NLP) techniques to deliver personalized course recommendations from large scale educational platforms.Course descrip-tions are processed in a structured pipeline of text processing which includes tokenization,stopword removal and lemmatization to get more readable and clear representation of course content.To know the similarity between a course and interest of a learner these descriptions are converted to numerical form by using TF-IDF vectorization and cosine similarity is used.We employ a hybrid scoring system that combines course ratings and similarity scores in a balanced way to further improve the output,considering both quality and rel-evance.**

Many experiments were performed on dataset gathered from Coursera which contains course descriptions,difficulty level,institutional affiliation,skill tags and learner ratings.The system's has the capability in recommending relevant courses across different domains which include A1,Data Science,Machine Learning and Web development.The recommendations are also presented visually,which aloows end users to understand the scores more easily.The whole system is lightweight and scalable and a useful tool for individual educational advice.Future directions for improving recommendations include combination of collaborative filtering with deep learning architectures and the integration of real-time feedback.

I. INTRODUCTION

Online education has transformed a lot in last ten years.It has changed the way education and skill is accepted.There are various platforms such as Coursera,Udemy that ofeer course.These courses are developed to cover all kinds of subjects so learner can study in his own pace.However it also comes across as a challenge when there are so many choices that is difficult to choose.Students face difficulty finding courses that match their background,interests and goals.Many of them fail to make good decisions in spite of spending hours looking.

This is not something that is a small problem.Poor course selection leads to waste of time and late development of skills. The gap between what students want to learn and what they can learn is widened because the course catalog is increasing on daily basis. To solve this problem and understand the intention of learner a hybrid course recommendation was proposed where Natural Langauage Processing is used to obtain key content features from the course descritions.Then these features are transformed into a numerical format by TF-IDF vectorization and directly compared with user's query by cosine similarity.These similarity scores are used in a hybrid scoring system,together with course ratings such that recommendations reflect not only on topic relevance but also the overall quality of the course.This system makes sure that it makes practical and reproducible in order to provide personalized education with the help of python implementation and evaluated Cousera dataset.

II. PROBLEM STATEMENT

Today's online learning is contradictory to the boom in online education.The number of courses is growing but choosing the right course is still a difficult challenge.Most users look immediately at title and detail description of course and select a course that look decent rather than one that matches their skills,background and desired learning.The trial-and-error process is time consuming and it will not always lead to successful matches,therefore it prevents learning progress rather than facilitate it.The basic problem lies in the absence of a "smart" and "guided" way of choosing what a student really needs.Most platforms just return search results about words entered but they don't know what courses are searched or what courses are good. As a reult,the lack of content for the students with different interests and different content requirement appears.The aim of this project is to fill these gaps.In this paper,we present a course recommendation system that is based on NLP techniques for understanding and connecting descriptions of courses and user input.

In other words,this system uses "semantic understanding"of things rather than word matching.In order to reduce the learner's work the system suggests courses based on similarity of content and course ratings.

III. OBJECTIVE

The overarching aim of this work is to build a functional and intelligent system capable of matching learners with courses that genuinely align with their interests and learning requirements.Instead of matching keywords and phrases,it employs Natural language processing and Machine learning techniques to understand the meaning behind course material and user queries.The research has the following specific goals,in an effort to achieve the above broader one: to achieve the above broader one:

- Text Preprocessing:Course description cleaning and normalization by means of NLP techniques including tokenization,removing stopwords,lemmatization and preserving meaningful information for further analysis.The course descriptions are vectorized by means of TF-IDF,in order for the system to calculate and compare the course contents computationally effective.
- Relevance Measurement:It helps measure the closeness of the learner's stated interests to the course content based on cosine similarity method and ranking similarity
- Hybrid Scoring Integration:Course rating information that already exists is incorporated into content-based similarity scores for creating an integrated scoring mechanism..
- Personalized Course Recommendation:The findings from the processes listed above are utilized to generate a list of course recommendations on contextual relevance

IV. LITERATURE REVIEW

The widespread use of online learning platforms sparked a significant academic attention in aiding learners navigating the ever-growing range of courses.Early education websites commenced to deal with this with the most straightforward search options and category browsing.While these mechanism allowed the learner to start,while the entire task of discovering left to the user.It would take a considerable amount of time and effort for the learner to go through the options before finding ideally matching ones.Content-based filtering is one of the most popular approaches for suggesting ideas.This technique examines information about a particular course,including its description and topics,as well as skills and other elements,and makes recommendations to users.In this case,the use of TF-IDF vecrorization and cosine similarity are essential techniques.These tools can be useful by assessing so a user's interests can be appropriate for the content.Both content-based and collaborative filtering approaches are becoming more popular.Some of the behavioral metrics,among enrollment,rating,or platform ineraction,have been added to enhance the recommendation process.The individual users can now suggest courses and become social.The major drawback is requiring a lot of historical data.

One of the major challenges in recommendation sysytems is the cold-start problem,which arises when there is insufficient information about new students.To grasp the limitations,scholars began to investigate hybrid recommendation methods.In hybrid systems, there are several signals rather than a single source of information.They consider the relevance of the content and rating of peers as well as the characteristic of their course to provide a more reliable and balanced recommendation.

Research show that the integration of content-based and collaborative can lead to improved performance in recommending products and improve the satisfaction of learners.Tokenization,stopword removal and lemmatization were done before the course could be processed,allowing for more meaningful features to be extracted from unstructured text rather than just simple keyword.leads to a better understanding of the purpose of its content.
In this field there has been some progress but the literature is still limited and does not treat the problem in systematic way.Most of the recommendation system that are present are built on content similarity or user behaviour instead of a proper combination of both.This gap is an impetus towards the present work,which aims to develop a hybride methodology combining the complementary perspectives and making them part of a unified recommendation framework.

V. METHODOLOGY

This part outlines the process taken in designing the pro-posed Hybrid Course Recommendation System. The course descriptions are taken into account by the recommendation engine. The final goal is that courses need to be determined in consultation with and aligned with user interests. The application of NLP techniques is crucial when it comes to improving the user experience of an application. The process utilizes techniques such as TF-IDF and cosine similarity in measuring the similarity between the user query and course data.

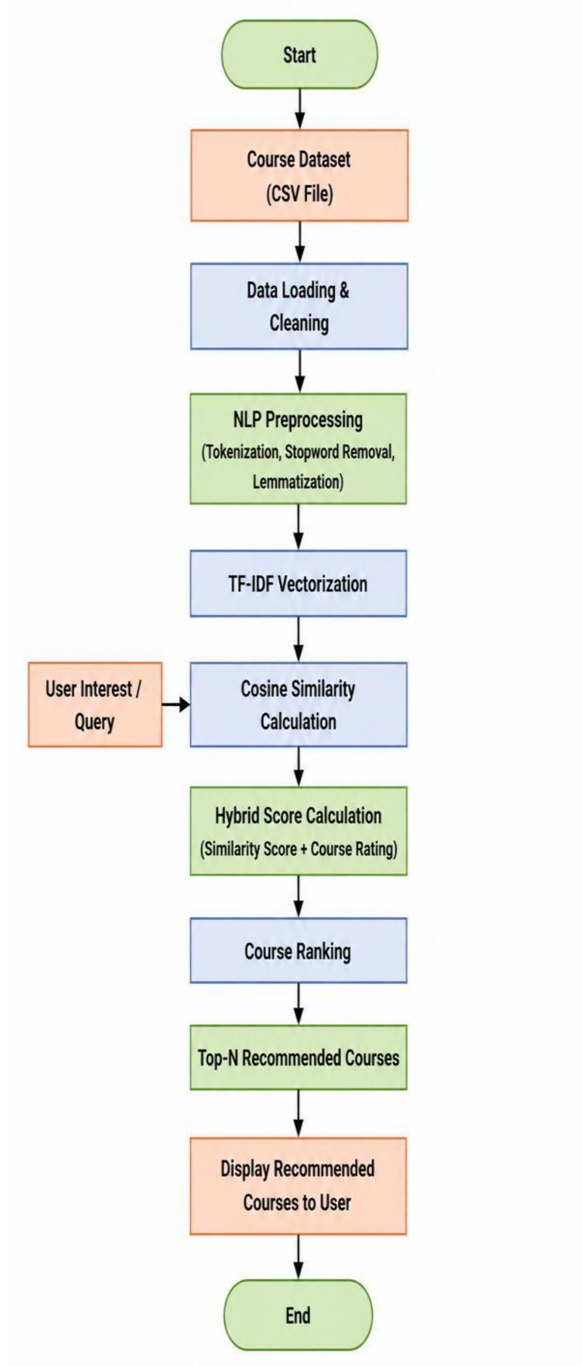

Fig. 1. Workflow of Hybrid Course Recommendation System

The first step towards recommending courses involves the loading of the course dataset and preprocessing of the dataset for analysis. The dataset involves a large amount of text data. Therefore, it is necessary to carry out cleaning and preprocessing on the dataset before recommendations can be generated. Descriptions of the course dataset are preprocessed using NLP algorithms like tokenization, stopword removal, and lemmatization. The cleaned text is then mapped into numeric values by using the Term Frequency-Inverse Docu-ment Frequency method. The TF-IDF method determines how important a particular word is in relation to other words in the dataset as shown below:

$$\text{TF-IDF}(t, d) = \text{TF}(t, d) \times \text{IDF}(t) \quad (1)$$

where the inverse document frequency component is defined as:

$$\text{IDF}(t) = \log \frac{N}{df(t)} \quad (2)$$

Here, TF(t, d) refers to the frequency of term t in document d, N stands for the total number of documents, and df (t) refers to the number of documents in which term t appears.

These TF-IDF vectors thus obtained are further used to calculate the cosine similarity between user interest and course descriptions. This helps in finding the relevant courses in respect to the learner's query and can be defined as:

Cosine Similarity(A,B) = $A{\cdot}B\ \|A\|{\times}\|B\|$ (3)

where A is the vector representing the user query, while B is the vector representing the course description. In order to increase the accuracy of the recommendation, a hybrid scoring model is used that incorporates similarity scores and course rating. The recommended score for the recommendation is computed as:

Hybrid Score = 0.7 × Similarity Score + 0.3 × Course Rating (4)

Recommendations made using such an approach take into consideration not only the relevance of the contents but also quality and popularity of the courses.

VI. IMPLEMENTATION

All the fuctionality was provided in Python,which is a suitable language for this application as it has compability with various kinds of libraries used for data management,text manipulation and machine learning tasks.Data was pulled from the Corsera website and organized it using Pandas,making it simple to manipulate.Titles,descriptions,grades and difficulty of the course materials are arranged.The Natural Language Tollkit(NLTK) was utilized to preovide basic implementation to obtain the necessary NLP functionality for text-baqsed processing.The complete course materials for computational analysis are now available.

A. Text Preprocessing Pipeline

Pipeline is used to process the raw course descriptions for text pre-processing ,following a number of steps for analysis.The system was first designed to manage de-scriptions by breaking down them to their samallest form.A large list of high frequency words which contribute little in terms of meaningful semantic information in a course was removed(e.g.articles,prepositions and conjuctions).That's the same list.These words are known as stopwords and can be removed which were in alphabetical order of the tokens.The use of comparision allows the words to be grouped and treated as equivalents,and facilitates cxlassification by tracing their differences.The descriptions was vectorized into numeric vector form using TF-IDF vectorization.The significance of the words in the particular description is hightened but not in that course.The question was preprocessed and projected to the vector space to get the topical similarity by means of cosine similarities using the same pipeline as the course descriptions.The cosine similarity does not care about the size of the vectors,only with regard to their direction.

B. Hybrid Scoring and Final Recommendations

Comparative is valuable but not enough to tell us what past students thought of a course.Why?We have incorporated a hybrid scoring system that includes course ratings into the final score.ore accounts for 70% of the composite score for each course and the rating for normalized courses accounts for 30%.Then the courses were ranked using reverse hybrid score quality-based filtering not only offers more relevant suggestions,but also enhances the likelihood of an enjoyable learning experience.The results of the implementation clearly demon-strate that the use of content-based filtering together with the use of the rating-aware scores gives a more reliable and user-centric recommendation result than the seperate use of either approach.

VII. RESULTS AND DISCUSSION

The proposed hybrid recommendation system was tested on the Coursera dataset which includes a wide range of course entries with attributes including: course name, full description, ratings by learners, offering institutions, and difficulty classi-fication. To guarantee that only semantically meaningful con-tent provided input to the similarity computations, all course descriptions were fed into an established NLP preprocessing pipeline, including tokenization, removal of stopwords and lemmatization, before making any recommendations.

A query representing a typical learner's search, such as "machine learning python ai" was used to test the system in a realistic manner, mimicking the informal, interest-driven nature of queries that a user might enter. The system has been able to identify and rank courses which are closely related to the domains of Artificial Intelligence and Machine Learning. Of all the courses examined, *AI for Everyone* has the highest hybrid score of 0.711, closely followed by *Introduction to Artificial Intelligence (AI)* with a score of 0.709. The difference between the two scores is so fine that either could be a strong match for the given query, perhaps with slight variations in content match or learner rating contribution. The overall high uniformity of scores among top recommendations shows that the top courses that were shortlisted were not only a good match, but they were in fact a good match overall, with each course receiving a fairly similar score, reflecting the hybrid scoring process in action.

	course_name	course_rating	difficulty_level	Hybrid Score
2175	AI For Everyone	4.8	Conversant	0.711333
2884	Introduction to Artificial Intelligence (AI)	4.7	Conversant	0.708927
2882	Introduction to Artificial Intelligence (AI)	4.7	Conversant	0.708927
2573	Introduction to Artificial Intelligence (AI)	4.7	Conversant	0.708927
2883	Introduction to Artificial Intelligence (AI)	4.7	Conversant	0.708927

Fig 2. Top Recommended Courses

The recommendation scores were presented graphically to support clearer interpretation of ranking outcomes. As illustrated in Fig. 3, the bar chart of hybrid scores reveals tight clustering among the top-ranked courses, reinforcing the observation that the recommended results were of broadly equivalent relevance to the input query. This visual output serves a practical purpose, allowing learners to quickly assess not just which course ranked highest but also how compa-rable the alternatives are, thereby supporting more informed decision-making.

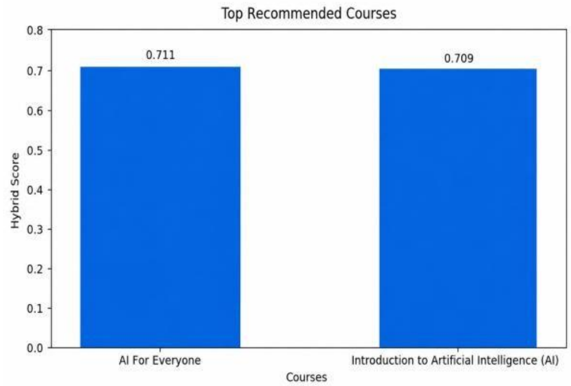

Fig. 3. Top Recommended Courses Based on Hybrid Score

Fig. 2 further highlights individual course scores, with *AI for Everyone* registering 0.711 and *Introduction to Artificial Intelligence (AI)* scoring 0.709, confirming that rankings are driven by genuine content alignment rather than disproportionate influence from either scoring component.

As shown in Fig. 3, the hybrid system demonstrates a clear advantage over single-method approaches by drawing on multiple complementary signals. Pure content-based systems risk recommending courses that are topically relevant but poorly received by learners, while rating-only approaches may surface popular courses that do not align with a specific user's interests. The proposed system addresses both limitations by integrating NLP preprocessing, TF-IDF vectorization, cosine similarity measurement, and course rating data into a unified scoring pipeline. Overall, the experimental outcomes confirm that the devel oped system is capable of delivering meaningful, personalized course recommendations, equipping learners with a reliable mechanism for identifying courses that genuinely match their goals and reducing the cognitive burden associated with man ual catalog browsing.

VIII. CONCLUSION

The hybrid Course Recommendation System was designed to overcome the problem associated with picking out the suitable courses from the vast amount of courses available over the web. The current research endeavors to solve the real-time problem of assisting learners to choose a course of interest out of massive amounts of data of online courses. Online learning courses allow your users to select the courses that best match their goals, objectives and background. This is a challenging problem and a solution is proposed to it, which can be implemented on a Natural Language Processing platform called Hybrid Course Recommendation. Our hybrid course Recommendation system uses Natural Language Processing techniques, rather than trying to do it in a general, but inef fective manner. We use a NLP pipeline to measure the exact semantic similarity between the individual user's interests and the specific courses offered, by analysing the text in the course descriptions, computing the TF-IDF of the same. A weighted combination of the content relevancy (measured by the cosinesimilarity function) and the learner feedback ensures that content relevancy is accorded the appropriate weight against the quality of the courses for the relevant ranking of the courses.

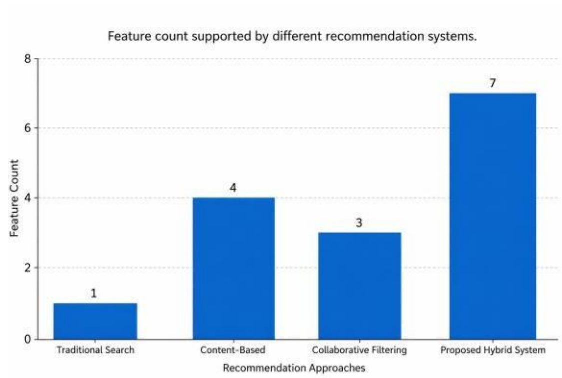

Fig. 4. Comparison of Existing Recommendation Approaches and the Proposed Hybrid System

Experiments show that the hybrid model outperforms single recommenders and high hits in the top ranked recommendations are also given a high hybrid score which is useful to satisfy real user demands. The proposed system not only enhances the academic part of it but also gives an advantage to the learners in choosing relevant courses and reduction of time. The system is also portable, interpretable and light weight, which enables the implementation of this system in real world platform. Some future works that could be helpful in this system are:1. A Collaborative filtering aspect can be incorporated so as to benefit from collective behaviour of similar learners. Use of more advanced deep learning based models like BERT/ sentence transformers so that a better semantic representation can be used. It could also be beneficial to use real time user feedback and update for real world applications.

IX. FUTURE ENHANCEMENTS

The existing system provides recommendation for content from the course, based on the average rating from students. This system for recommendations is fine, and reliable. How-ever, there are a number of improvements that can be done in future, that would greatly increase the capacity of the system.

1) Collaborative filtering There is one, particularly powerful improvement to make: incorporating collaborative filtering techniques. For now, only the attributes of the course are used in the system, and there is no awareness of student behavior. Adding a layer of social intelligence to the recommendation engine, in the form of how similar students interacted with, liked and took the course, is very powerful for generating recommendations which more personally relate to the needs of the student; they can know how likely it is that the learner will be interested in some material, even if the learner cannot say how or why they want to know it. Another advance-ment is related to advanced Natural Language Processing technologies,

2). The current TF-IDF based representation of content could be the simple counting method and it has the potential to ignore the content of the text and other deeper lingustic concepts, which would be advantageous for the next iteration of the recommender system using recent advances in the field of NLP. More generally, a switch to models that convey semantic and deep-linguistic relationships among textual inputs like BERT, or just to add transformer based models to such system, would enable to infer the real meaning of the interactions and questions posed by student to student.

ACKNOWLEDGEMENT

It is a matter of great pride for me to thank my guide for all the help, guidance, and encouragement that he has provided while working on this project. I would like to thank my college and department for provid ing the necessary facilities to perform this project effectively. Thank you to all the websites and research papers that were used in order to acquire information about Natural Language Processing and recommendation systems. Scikit-learn, NLTK, Pandas, and the Coursera Dataset are some of the sources that helped during this project. Last but not least, I would like to extend my thanks to my friends for their continuous encouragement and support during the completion of this project.

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, Jun. 2005.
[2] F. Ricci, L. Rokach, and B. Shapira, Recommender Systems Handbook, 3rd ed. Cham, Switzerland: Springer, 2022.
[3] Y. Koren, R. Bell, and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," IEEE Computer, vol. 42, no. 8, pp. 30–37, Aug. 2009.
[4] G. Salton and C. Buckley, "Term-Weighting Approaches in Automatic Text Retrieval," Information Processing & Management, vol. 24, no. 5, pp. 513–523, 1988.
[5] K. Sparck Jones, "A Statistical Interpretation of Term Specificity and Its Application in Retrieval," Journal of Documentation, vol. 28, no. 1, pp. 11–21, 1972.
[6] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and Their Compositionality," in Proc. Advances in Neural Information Processing Systems (NeurIPS), 2013, pp. 3111–3119.
[7] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proc. NAACL-HLT, 2019, pp. 4171–4186.
[8] C. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval. Cambridge, U.K.: Cambridge University Press, 2008.
[9] S. Bird, E. Klein, and E. Loper, Natural Language Processing with Python. Sebastopol, CA, USA: O'Reilly Media, 2009.
[10] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.
[11] W. McKinney, "Data Structures for Statistical Computing in Python," in Proc. 9th Python in Science Conference (SciPy), 2010, pp. 56–61.
[12] J. D. Hunter, "Matplotlib: A 2D Graphics Environment," Computing in Science & Engineering, vol. 9, no. 3, pp. 90–95, 2007.
[13] Coursera Inc., "Coursera Dataset," Available: https://www.coursera.org/. Accessed: Aug. 2026.
[14] M. F. Porter, "An Algorithm for Suffix Stripping," Program, vol. 14, no. 3, pp. 130–137, 1980.
[15] T. Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features," in Proc. European Conference on Machine Learning (ECML), 1998, pp. 137–142.
[16] P. Resnick and H. R. Varian, "Recommender Systems," Communications of the ACM, vol. 40, no. 3, pp. 56–58, Mar. 1997.
[17] X. Su and T. M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," Advances in Artificial Intelligence, vol. 2009, Article ID 421425, 2009.
[18] D. Jurafsky and J. H. Martin, Speech and Language Processing, 3rd ed. (Draft). Pearson, 2023.