International Journal of Computer Sciences and Engineering
Vol.**11**, Issue.**9**, pp.**28-34**, September **2023**
ISSN: 2347-2693 (Online)
Available online at: www.ijcseonline.org

Research Paper

A Collaborative Student Course Recommender System for Learning Analytics

Nonye Emmanuel Maidoh[1]

[1]Dept. of computer science, AkanuIbiam Federal Polytechnic, Unwana, Ebonyi State, Nigeria

Author's Mail Id: nonye.maidoh@yahoo.com

Received: 26/Jul/2023; **Accepted**: 30/Aug/2023; **Published**: 30/Sept/2023. **DOI**: https://doi.org/10.26438/ijcse/v11i9.2834

Abstract: Assessment of learning outcomes among students at institutions of higher education is a fascinating issue with a wide range of possible applications for all parties including students, administrators, potential employers, etc. Even while education is becoming more widely available and popular, the high drop-out rates remain a challenging issue. Choosing the best courses for your area of specialization can be difficult and time-consuming. The majority of the current course selection algorithms do not consider courses that match student talents, the user's future professional goals, or the user's ideal job based on such objectives. The goal is to create a powerful learning analytics system that can effectively recommend courses to students based on their preferences and skills. The collaborative filtering recommender (CF) approach, which combines KNN and decision tree approaches, was used to match courses, abilities, requirements, and interests with recommended lists. The effect of user skills on the recommendation platform was investigated by altering a number of suggestion quality features. A collaborative filtering recommender system was developed by fusing KNN and DT to suggest specialized courses for college students. This improved the standard of the recommendation system. A recommender model was constructed with cosine similarity matrix of student course descriptions in order to include new descriptions that are being suggested to the overall descriptions. Term frequency inverse document frequency was used to convert the entire course description into a vectored representation of words.

Keyword: Collaborator filtering, decision tree, KNN, machine learning, recommender system

1. Introduction

The use of recommendation systems (RSs) in a variety of applications has increased their popularity over time[1]. This also applies to the choices that institutions make about the courses they offer to students. It has been extremely successful to predict users' quick response to recommendations using collaborative filtering[2] (and supervised learning-based techniques[3],[4]. The following three main methods for recommendation systems can be distinguished: collaborative, content-based and hybrid filtering recommender systems. Collaborative information filtering techniques are used by recommender systems to process data and present the user with possibly more pertinent items. The term "Item" is a broad one that refers to the content that the system suggests to the consumers. It might be a book, movie, or item of product. All individuals with a prospective need for assistance in evaluating the alternative things that an information system may provide are the target users of RSs. An artificial intelligence(AI) program known as a recommender system that uses big data to advise or recommend products/items to clients. This AI algorithm is typically connected to machine learning(ML). The demand for recommendation systems grows along with the rate of data generation. Users judge the quality of recommendations based on factors other than their correctness, necessitating a willingness to explore data[5],[6]. A colossal amount of information surrounds us as we approach the age of information and communication systems[7]. Intelligent decisions are made when there is more details available, yet occasionally too much knowledge can lead to poor decision-making. Users are given individualized choices for items through systems that suggest items that they may find interesting[8**].** The phrase "item" refers to whatever the system suggests to the user. Tan *et al.*[9*]* in order to increase user exploration in recommender systems put in place a content-based recommendation system that takes into account the commonalities between the course contents[10]. There hasn't been any effort to provide a user-specific course ranking list that centered on user interest and requirements.

The existing course recommendation algorithms do not take into account student area of specialization that matches students skill set, the user's future career ambition, or the user's desired job. Instead, they recommend programs from a communal perspective[11]. Finding areas of specialization for students using their respective skills can be a difficult and time-consuming. Finding the best places to check for course offers, the specifics and substance of each course, timeframe, requirements, lecturers, workload, etc. are all important considerations when selecting a course and area of specialization. The aim is to develop a collaborative Student Course Recommender using KNN and DT techniques for Learning Analytics. The impact of user skills on recommendation platform was examined by changing several features of recommendation quality.The exploration research in DT and KNN techniques served as the inspiration for the set of tactics that were provided. Machine learning techniques were used to create a collaborative filtering recommender system that takes user preferences and talents into account while reducing information overload. With the right dataset, this was set up to train and test the suggested system. It is proposed that students in higher education can use a collaborative filtering recommender system that combines KNN and DT to suggest courses with areas of specialization. This is done to raise the quality of recommendations.

This paper is organized as follows: Section 1 provides an introduction; Section 2 offers a brief evaluation of prior approaches related to the topic and the gap in studying the proposed model; Section 3 introduces the model's materials and methods; Section 4 covers the results and in-depth discussion of the results; and Section 5 provides the paper's conclusion.

2. Related works

A system of recommendation is a machine learning technique that is automated and originated in order to analyze or extract certain useful items such as songs, people, products, and etc. Building a recommender system involves three main stages: gathering data, gaining knowledge and recommendation[12]. There are numerous methods used to create recommender systems, such as collaborative, content-based filtering, and hybrid systems for recommendation.

(a). Collaborative Filtering: Collaborative filtering is a widely used method for creating recommender systems. It is based on gathering and studying a lot of data regarding users' actions, interests, or behaviors in order to predict what users would like dependent on how they are similar to other users[13]. CF's ability to accurately propose complicated items like movies without requiring an "understanding" of the item itself is one of its main advantages because it does not rely on content that can be machine-analyzed. It doesn't take a professional to provide recommendations using collaborative filtering because it keeps track of changes in user preferences over time when creating a model from a user's behavior. A distinction is frequently drawn between explicit and implicit kinds of data gathering. It creates a problem with recommendations that aren't very diverse due to its tendency to suggest well-liked things with high ratings[14].

(b). Content-based filtering: Content-based filtering(CBF) is another strategy that is frequently used when constructing recommender systems. The foundation of content-based filtering techniques is the item's description and the user's profile of preferences[15]. In a content-oriented recommender system, items of interest are described using keywords, and the user's preferences are shown in their profile. In simple terms, these algorithms try to recommend items that are comparable to ones that a user has previously liked. The user's prior ratings of various potential items are specifically compared, and the best-matching items are then recommended. Research on information filtering and retrieval is the basis of this strategy. Content-based recommenders approach recommendations as a user-specific classification issue and learn a classifier for the user's preferences based on the characteristics of the items being recommended. It is simpler to deploy and much more scalable because the content-based model does not need information about the preferences of other users[16]. It lacks creativity and variation in its predictions despite concentrating on similar qualities and finding it difficult to promote novel products that have not yet been assessed by several people.

(c). Hybrid Recommender Systems: There are numerous ways of building hybrid approaches: by making content-based and collaborative-based predictions, then integrating them; by giving a collaborative-based capabilities (and vice versa); or by blending the two approaches into one. It can improve the precision and dependability of recommendations, ease the constraints placed on user recommendation techniques, and enable more customized recommendations[17].

Chang.*[18]* used a user-based CF algorithm in order to create a course recommendation system using expected course grades as the criterion to rate potential courses. Students' courses taken, grades earned, and popularity of teachers during a five-year period are all part of the data set, and clustering was done using the artificial immune system concept.With the goal function being the antigen and the potential solutions being the antibodies, this concept offers a method for determining the optimum answer to an optimization problem. The students are then grouped based on the outcomes, and user-based collaborative filtering is then used to anticipate how well the courses will be rated and produce recommendations. Volkovs& Yu [19] described how to create effective collaborative filtering (CF) models when there is just binary feedback available. To guide latent factorization and generate latent representations, the concept of neighborhood similarity was introduced. It was done by using matrix factorization techniques like Singular Value Decomposition (SVD) to improve the performance of CF modelsThis method outperformed the standard CF techniques on binary feedback data sources when evaluated on multiple sizable public datasets. A system was developed by Khoja&Shetty[20] with the purpose of recommending courses. A collaborative filtering strategy was presented by Dwivedi&Roshni[21] in recommending elective courses to students. This was determined by computing the item similarity between courses depending on the student's achievement in several courses.Bokde*et al.[22]* applied dimensionality reduction techniques with a collaborative filtering strategy based on student performances for several criteria. However, the study's lack of clarity regarding the data collection process makes it difficult to generalize the method. Hsu [23] investigated how to recommend individual learning activities in learning English and writing abilities measuring models in order to help systems choose which activity or assignment is suited for a particular student. Jena et al.[24] developed e-learning-based course recommender system for students in a pandemic condition. They provided the end user with several kinds of course selection options and asked them to choose the best one. This was created to offer users better and automated services while choosing their careers. They incorporated KNN, SVM, and ANN-based filtering mechanisms, among other machine learning approaches. This made it easier for users to choose online courses that sparked their curiosity. KNN was found to perform better than SVM and ANN methods.

3. Materials and methods

This chapter discusses the potential for recommending courses to students using the recommended DT and KNN technique. The proposed study approaches are offered and assessed in order to put the suggested architecture into practice

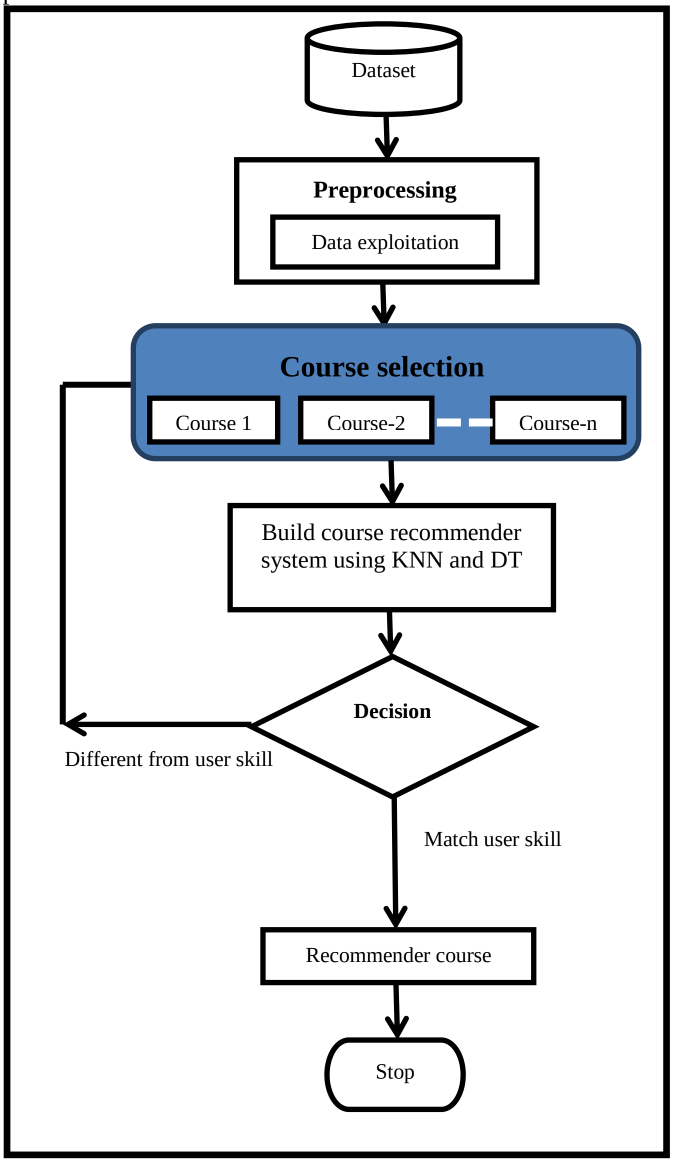

Figure 1: Architecture of proposed system

We are using a collaborative filtering recommender approach in combination with decision trees and KNN technique in order to match courses to student interests and requirements. A cosine similarity matrix of the student course description was created andnew course descriptions are being added and recommended. Following that, student descriptions are transformed using term frequency inverse document frequency into a vectored representation of words (TFIDF). For DT and KNN-based recommender systems, we suggest a set of measures for gauging the many aspects of suggestion quality, including accuracy, diversity and novelty.

Dataset: The dataset, which includes course id, gender, faculty, subjects, marks obtained, results columns for courses and specialization, will be collected from various schools, departments, and faculties(schools) throughout Nigeria.This will make it possible for us to create a content-based recommender system that can give students course recommendations based on their interest, knowledge and requirements. RS's implementation can be broken down into two steps:

(a). Pre-processing: Preprocessing is a set of procedures used to change our data before feeding it to the algorithm[25]. It changes data into a format that can be used for data mining, machine learning, and other data science operations more quickly and efficiently. Data set may therefore be incomplete, contain manual input errors, contain redundant data, or use several names to represent the same record. Data used to train machine learning algorithms must be systematically preprocessed because humans may frequently spot and fix these issues in the data they utilize in their line of work. In order to produce accurate, precise, and robust findings for industrial applications, ML and data science algorithm design require some sort of data pretreatment. Data integration, data reduction, data cleaning, and data transformation are the four processes that make up the data preparation step[26].

(b). Data Exploration: Data exploration(DE) naturally results in recommendations for things that are less relevant to the user's known interests, which is generally perceived as a cost to the user experience, especially in the short term[27]. If data contradicts the model's basic assumptions or contains flaws, we won't be able to utilize our ideal model to generate the desired results. If we don't engage in data exploration, then we can end up spending majority of our time checking models without identifying the problem in the dataset.

K-nearest Neighbor: The KNN requires no assumptions about data distribution in predicting target variables. It works based on observations about data features that are similar to the points in one particular class known as nearest neighbors[28]. The k parameter specifies number of neighboring points used to classify similar data points into one class with the concept of voting.The researcher created KNN classifier object and passed the k-neighbors argument value to a function in the K-neighbors Class. The KNN method is used to search for and locate the k-nearest neighbors on the basis of the cosine similarity value[29],[30].

The cosine similarity formula is given in equation 1 and is used in determining the angle between two rating vectors:

$$\text{Cos } \boldsymbol{\theta} = a.b/||a|| \times ||b|| \quad (1)$$

Whwere $\boldsymbol{\theta}$ depicts the angle between two rating vectors, a and b are the two rating vectors, a.b represent the dot product while ||a|| × ||b|| denote the cross product for the length of two vectors. User_id, talents, and the number of courses are among the input data fed into the student course recommendation system[31],[32].

Decision tree(DT): We employed the DT model to recommend students bases on their skills and preference using set of decisoon rules with fine-tuned hyperparameter values. The ranking of attributes are using the information gain with attribute importance.

The formula given in equation 2 is used as a measure to reduce variance in DT.

$$D = \frac{1}{\ell}\sum_{\ell=1}^{1}\left(y_i - \frac{1}{\ell}\sum_{j=1}^{\ell} y_j\right)^2 \quad (2)$$

Where ℓ = no. of items in DT leaf nodes and y_i is the target variable representing employee exirting or not leaving.

A DT class was created with some instances containing maximum depth set to 15 and random states to 0. The model was trained with (80%) of the total dataset, tested and predicted with the remaining 20% for validation purpose.The existing system worked on the voluntary leave; which is a situation whereby an employee decides to quit and leave the Organization on his/her own accord for some reason(s). The system is a product of comparison. A small data set was used with limited number of attributes and records which may have caused Imbalanced data. It gives a binary tree that uses information gain (entropy) as it's splitting criteria. Its pruning technique adopts the binomial confidence limit method. In a case of handling missing values, C5.0 allows whether estimate missing values as a function of other attributes or apportions the case statistically among the results.

(c). **Learning process** that learns the relationships between objects to create a model that depicts a user's interests or skills[33],[34].

(d). **Decision process**: As part of the decision-making process, the user's preferences and restrictions are taken into account, and the model created in the preceding stage is used to produce predictions of the courses that are suitable for the user.

Algorithm :Course Recommendation system

Step	**Processes involved**
Input	Course dataset
Output	Course recommender system
1	Begin
2	Make choice from available courses according to priority
3	Build a recommendation on variety of courses
4	Define a recommendation function
5	Create a similarity matrix between students
6	Get pairwise similarity scores of all students
7	Sort the course descriptions based on similarity level
8	Recommend courses to students
9	End recommendation system
10	Return

Considerations are made using dataset from the course recommender system that includes the course_id, course title, user id, price, level and rating details. Curses and users' unique course index numbers were calculated. Lists of recommended and rated courses were made available on the output screen and a utility matrix was produced for the course data that is made visible. The maximum number of rows in the empty array is set to the total number of courses. The computed matrix was applied to the hybrid model.A dictionary is built to acquire the estimated index and matrix and integrate them in the hybrid model. A procedure

4. Results And Discussion

Decision tree and KNN-based recommendation systems fared the best according to the results. The proposed hybrid technique was able to recommend courses in a particular area of competence based on user skill and knowledge,It became easier to search for someone with expertise with comparable skill features.

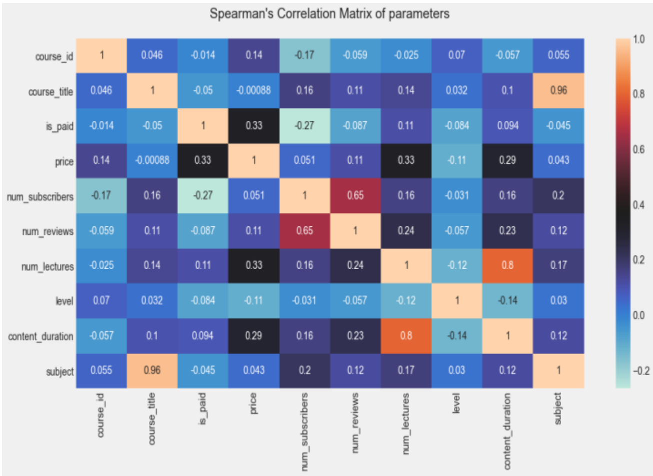

Figure 2: The speaman's correlation matrix of parameters

Figure 2 shows the feature correlation matrix of the dataset for our proposed system, emphasizing the clusters of highly correlated features that significantly influence the recommendations. It shows graphical representations of related features and pairwise cosine similarity.

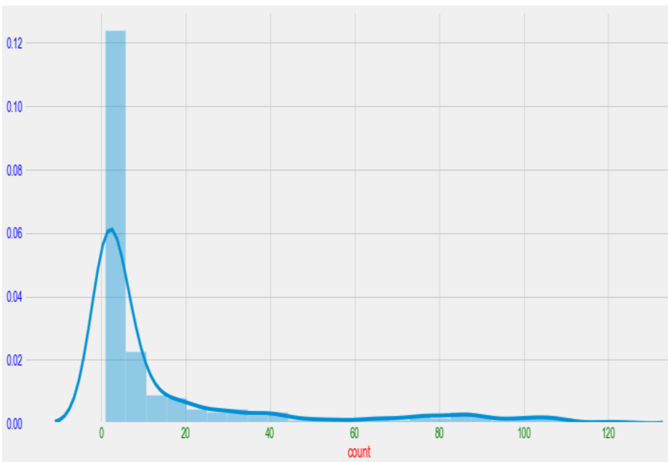

Figure 3: Distribution lecturers across courses

Figure 3 shows how lecturers are distributed throughout all the different courses. When mapped with courses, the distributions of lecturers to courses are exceptionally high from 1 to 10 and decreases to the lowest number.

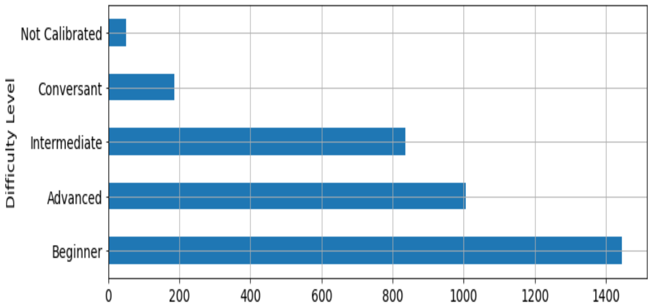

Figure 4: levels of courses

Figure 4 shows the various course levels, including beginner, intermediate, advanced, and not calibrated courses. The beginner's level, which requires no prior expertise and is simple to learn, is intended for individuals who are just getting started. The intermediate courses are intended for people who are skilled and passionate about a certain field who may want professional assistance with their technical and academic abilities. Students have the chance to receive university credits for taking courses in advanced subjects in higher education. Advanced, intermediate, conversational, and un-calibrated courses yielded the lowest results, with the beginners course producing the highest (around 1400).

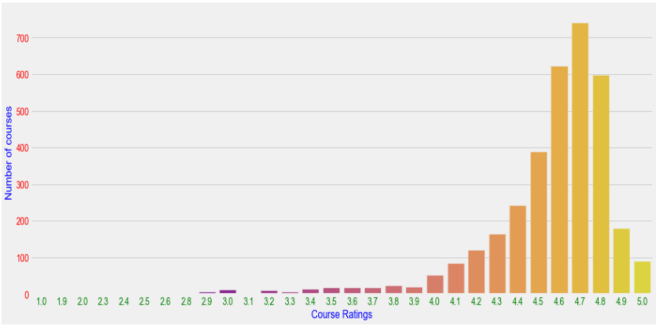

Figure 5: Course ratings

Figure 5 displays the rated course values together with their relative importance for making recommendations. The course recommendation algorithm signed the rating values for the most important courses with 4.7, 4.8, and 4.6. The rating values range from 1.0 to 5.0.

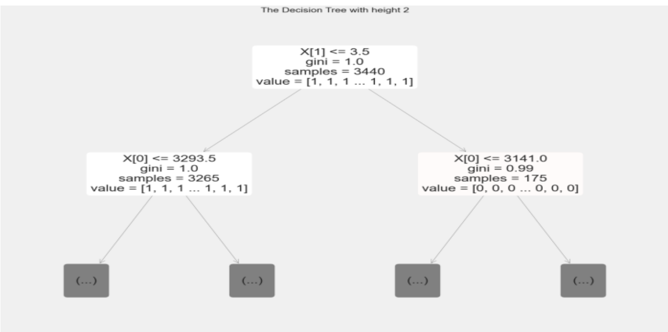

Figure 6: Decision tree

Figure 6 shows the decision tree for height-3 generated from the sample dataset for the proposed system with gini and target value. This made it easier to explain the logic of the model to the stakeholders because the tree generated in the figure is much simpler to understand for those without an ML background. The DT row values in each node provide details about the numerous observations which were sorted into a certain node and fall into each of the categories. One course could be completely separated from the others using the feature X[0], which is the tree length.

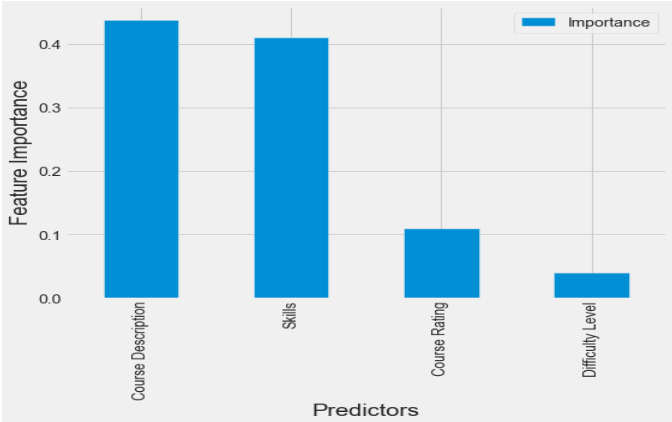

Figure 7: Most importance features

The feature significance plot is depicted in Figure 7 and shows the contribution of each feature to the model's ability to predict the desired values. The most important variables are the student's skills and the course description, followed by course rating and difficulty level.

Table 1: Those with programming skill

Rank	Recommended courses
1	Image Understanding with TensorFlow on GCP
2	End-to-End Machine Learning with TensorFlow on GCP
3	Building Resilient Streaming Analytics Systems on GCP
4	Smart Analytics, Machine Learning, and AI on GCP
5	Java Programming in recommendation System
6	Introduction to TensorFlow
7	Getting started with TensorFlow
8	Applying Machine Learning to your Data with GCP

Table 1 lists the recommended courses for people who are skilled in Python programming and have chosen a particular area of specialty based on their potential. The recommender suggested course that required programming skills usingTensorflow with GCP, End-to-end ML with tensorflow on GCP, machine learning and etc.

Table 2: Those with database skills

Rank	Recommended courses
1	Retrieve Data with Multiple-Table SQL Queries
2	Advanced SQL Retrieval Queries in SQLiteStudio
3	SQL for Data Science
4	SQL for Data Science Capstone Project
5	Manipulating Data with SQL
6	Analyzing Big Data with SQL
7	How to Design a Space-Saving Table Using SketchUp
8	Databases and SQL for Data Science

The recommended courses for those with database skills are listed in Table 2. The recommender offered a priority rating and suggested Retrieve Data using Multiple-Table SQL Queries and other options.

Table 3: Those with Mathematical skills

Rank	Recommended courses
1	Data and Health Indicators in Public Health Practice
2	Linear Regression in R for Public Health
3	Introduction to Statistics & Data Analysis in Public Health

The suggested courses for those with statistical and quantitative expertise are listed in Table 3. The recommender proposed three areas of specialization: Introduction to Statistics & Data Analysis in Public Health with a Priority Value, Data and Health Indicators in Public Health Practice, and Linear Regression in R for Public Health.

The weighted average of course ratings across every pair of tuples in the leaf node rating set was computed in order to create the course recommender system in an easy-to-understand way. This was utilized to recommend the course with the best weighted average in the area of expertise. The tree's leaf nodes retained user skill ratings according to each user's area of expertise as shown in figure 6. A better system was created whenever courses are chosen that have received the highest ratings from studentswhichare related to their skills. All of the recommended courses were determined on a weighted average since both courses and user skills can occur more than once. The normalized information gain criterion is used by the DT to select the attributes for dividing the tree nodes. The property with the most normalized information gain was chosen in order to achieve the most significant decrease in entropy.

Table 1 provides recommended courses for those who are proficient in Python programming. The system recommended courses that matched students' programming abilities, such as End-to-end ML with Tensorflow on GCP, machine learning, and others with ranked values that call for programming knowledge.A higher similarity level and course-related variables that completely matched user competencies were found in the students' recommended course.Students must have a basic understanding of Python programming libraries like Tensorflow, known as the "big daddy" of deep learning, in order to enroll in the recommended course.

Table **2** displays suitable programs for learning database programming. The recommender suggested Retrieve Data Using Multiple-Table SQL Queries and other methods, and provided a priority score.The system recommended courses based on the students' proficiency in database programming, including: databases for data science; SQL database programming skills that can be applied in data science capstone projects; data manipulation in SQL; Big data; designing space-saving database tables using Sketch-up.

Figure 3 depicts recommended courses for students having statistical and mathematical skills.The recommender offered the following subjects for students ought to specialize on: Data and Health Indicators in Public Health Practice, Linear Regression in R for Public Health, and Introduction to Statistics & Data Analysis in Public Health with a Priority Value.There was a substantial correlation between the recommended courses and student abilities or competency as observed in the tables above.

5. Conclusion

A recommender system that combines KNN and DT was developed to match courses to students' areas of specialization.The suggested courses were appropriate for the students' abilities in each area of study.The findings reported above show that there was a significant correlation between the recommended courses and student aptitude or ability.The majority of students prefer to focus their career decisions on a field of specialty based on their skill set. A course recommender system was developed using a collaborative filtering approach, and degree of similarity computed between courses and their field of study of specialization using the cosine similarity. There existed a significant relationship between the recommended courses with areas of specialization and student skill.The recommended courses precisely corresponded to the students' skill set.It was tested effectively using a collaborative filtering for course ratings which does not affect the predictions of the hybrid model. This could tremendously aid students in choosing better careers in their fields of study and assist devoted school officials in assisting students.

It's conceivable that new courses with feature-specific areas of focus can replace certain existing courses. Additional data can be gathered using other recommendation strategies as well as further similarity computation methods are required in order to improve the suggested recommendation system in subsequent studies.

Funding source

Tertiary Education Trust Fund of Nigeria (TETFund) provided funding for this study.

Acknowledgement

I amgrateful that this research was supported by an industrial-based research (IBR) grant from the Nigerian Tertiary Education Trust Fund (TETFund).The sponsor had no involvement regarding the design, collection of data and application of those findings. I would especially want to thank Dr. Felix Attah, the rector of AkanuIbiam Federal Polytechnic, Unwana, for approving and releasing the funding necessary to complete this task.

References

[1] S. Ziweritin, SC. C. Ukegbu, T. A. Oyeniran, I. O. Ulu, "A Recommendation Engine to Estimate Housing Values in Real Estate Property Market," International Journal of Scientific Research in Research in Computer Science and Engineering(IJSRCS), Vol.**9**, Issue.**1**, pp.01-07, **2021**.

[2]. Y, Koren, R, Bell, C, Volinsky, "Matrix Factorization Techniques for Recommender Systems," *Computer* Vol.**42**, Issue.**8**, pp.3-30, 2009.

[3]. P.\, Covington, A. Adams, E, Sargin, "Deep neural networks for youtube recommendations," In *Proceedings of the 10th ACM Conference on Recommender Systems*. ACM, pp.**191–198**, **2016**

[4]. S, Zhang, L. Yao, A, Sun, Y, Tay, (2019) Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys (CSUR)* Vol.**52**, Issue.**1**, pp**.1–38, 2019**.

[5]. Y, Zhang, F, Feng, X, He, T, Wei, C, Song, G, Ling, Y, Zhang, "Causal Intervention for Leveraging Popularity Bias in Recommendation," arXiv:2105.06067 [cs] [Preprint]., **2021**

[6]. K. Patel, H. B. Patel, "A state-of-the-art survey on recommendation system and prospective extensions," Computers and Electronics in Agriculture, Vol.**178**, pp.**10**, **2020**.

[7]. G. Aceto, V. Persico, A. Pescapé, " Industry 4.0 and health: internet of things," big data, and cloud computing for healthcare 4.0. J Ind Inf Integr 18(October 2019), pp.**100-129, 2020.**

[8]. A. Simeone, Y. Zeng, A. Caggiano,."Intelligent decision-making support system for* manufacturing solution recommendation in a cloud framework," The International Journal of Advanced Manufacturing Technology, Vol.**112**, pp.**1035–1050, 2021**.

[9]. H. Tan, J. Guo, Y. Li, "E-learning Recommendation System. In Computer Science and Software Engineering," 2008 International Conference on IEEE Hubie, Vol.**5**, pp.**430–433, 2008**

[10]. K. I. Ghauth, N. A. Abdullah, "Learning Materials Recommendation using Good Learners Ratings and Content-Based Filtering," Educational Technology Research and Development, Vol.**8**, Issue.**6**, pp.**711–727**, **2020**.

[11]. X. Li, S. K. Chang, "A Personalized E-learning System Based on user Profile Constructed using Information Fusion, Proceeding of the 11th International Conference on Distributed Multimedia Systems, pp.109–114, **2005**

[12]. S. Milano, M. Taddeo, L. Floridi, "Recommender systems and their ethical challenges," AI Soc. Vol.**35**, Issue.**4**, pp.957–967, **2020**

[13]. R. R. Pedersen, "Datafication and the push for ubiquitous listening in music streaming," MedieKultur: Journal of media and communication research, Vol.**36**, Issue.**69**, pp.**71-89**, **2020**

[14]. L. Porcaro, C. Castillo, E. Gómez, "Diversity by Design in Music Recommender Systems," Transactions of the International Society for Music Information Retrieval, Vol.**4**, Issue.**1**, pp.**114-126**, **2021**

[15]. D. Sánchez-Moreno, V. López Batista, M. D. M. Vicente, A. L. Sánchez-Lázaro, M. N. Moreno-García, "Exploiting the User Social Context to Address Neighborhood Bias in Collaborative Filtering Music Recommender Systems', Information, Vol.**11**, Issue.**9**, pp**.439**,**2020**

[16]. Q. Villermet, J. Poiroux, M. Moussallam, T. Louail, C. Roth, "Follow the guides: disentangling human and algorithmic curation in online music consumption,' in Fifteenth ACM Conference on Recommender Systems. RecSys '21: Fifteenth ACM Conference on Recommender Systems, Amsterdam Netherlands: ACM, pp.**380-389, 2021**.

[17]. T. Tofalvy, ,J. Koltai, "Splendid Isolation": The reproduction of music industry inequalities in Spotify's recommendation system," New Media & Society [Preprint]. pp.**1-20**, **2021**,

[18]. Z. Chen, " Intelligent Courses Recommendation System of Collaborative Filtering Algorithm Based on Kmeans Clustering under Spark Platform ," Advances in Social Science, Education and Humanities Research, Vol.**664**, pp.**2041-2045**, **2021**

[19]. M. Volkovs, W. Y. Yu, "Effective Latent Models for Binary Feedback in Recommender Systems," In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Vol.**4**, pp.**313–322**, **2015**

[20]. Z. Khoja, S. Shetty, "Hybrid recommender system for college courses," in *2017 International Conference on Computational Science and Computational Intelligence (CSCI)*, Las Vegas, NV: IEEE; pp.**1167–1171**, **2017**

[21]. S. Dwivedi, V. K. Roshni, "Recommender system for big data in education," in *2017 5th National Conference on E-Learning & E-Learning Technologies (ELELTECH)*, Hyderabad: IEEE, pp.**1–4, 2017**

[22]. D. K. Bokde S. Girase, D. Mukhopadhyay, "An Approach to a University Recommendation by Multi-criteria Collaborative Filtering and Dimensionality Reduction Techniques," in *2015 IEEE International Symposium on Nanoelectronic and Information Systems* (Indore: IEEE), pp.**231–236**, **2015.**

[23]. M. H. Hsu, "A Personalized English Learning Recommender System for ESL Students," *Expert Syst. Appl.* Vol.**34**, pp.**683–688**, **2008**.

[24]. K. K. Jena, S. K. Bhoi, ST. K. Malik, K. S. Sahoo, N. Z., Jhanjhi, S. Bhatia, F. Amsaad, "E-Learning Course Recommender System Using Collaborative Filtering Models," Vol.**12**, Issue.1, pp.**1-21**, **2023**

[25]. P. Covington, A. Adams, E. Sargin, "Deep neural networks for youtube recommendations," In *Proceedings of the 10th ACM Conference on Recommender Systems*. ACM, pp.**191–198**, **2016.**

26]. J. Marshall, H. Greenberg, P. A. Machun, "How would they choose? Online Student Preferences for Advance Course Information," Open Learning: The Journal of Open, Distance and e-Learning, Vol.**27**, Issue.**3**, pp.**249–263**, **2012**.

[27]. C. Pei-Chann, L. Cheng-Hui, C. Meng-Hui, ."A hybrid Course Recommendation System by Integrating Collaborative Filtering and Artificial Immune Systems. Algorithms,"Algorithms, Vol.**9**, Issue.**3**, pp.47, **2016**

[28]. R. H. Singh, S. Maurya, T. Tripathi, T. Narula, G. Srivastav, "Movie Recommendation System using Cosine Similarity and KNN," International Journal of Engineering and Advanced Technology (IJEAT), Vol.**9** Issue.**5**, pp.**556-559**, **2020**.

[29]. S. Jagtap, S. Borate, "Tourist Destination Recommendation System using Cosine Similarity," International Research Journal of Engineering and Technology (IRJET), Vol.**09**, Issue.**10**, pp.**106-110**, **2019**.

[30]. P. D. Nurfadilaa, A. P. Wibawaa, I. A. E. Zaeni, A. Nafalski, "Journal Classification Using Cosine Similarity Method on Title and Abstract with Frequency-Based Stopword Removal," International Journal Of Artificial Intelegence Research, Vol.**3**, Issue.**3**, pp.**28-37**, **2019-**

[31]. F. Fkih, "Similarity measures for Collaborative Filtering-based Recommender Systems: Review and experimental comparison," Journal of King Saud University – Computer and Information Sciences, Vol.**34**, Issue.**9**, **pp. 7645-7669**, **2022**

[32]. A. Agarwal, S. kumar, "Resume recommendation system using cosine similarity," International Research Journal of Modernization in Engineering Technology and Science, Vol.**04**, Issue.**4**, pp.**158-162**, **2022**.

[33]. V. L. Spandana, J. R. Abhinav, P. N. Sampath, M. M. Ahamed, "Course recommendation system using content based filtering and cosine similarity," Journal of Emerging Technologies and Innovative Research (JETIR), Vol.**10**, Issue.**3**, pp.**210-218**, **2023**.

[34]. L. R. Roopesh, B. Tulasi, "A Survey of Travel Recommender System," International Journal of Computer Sciences and Engineering(IJCSE), Vol.**7**, Issue.**3**, pp.**356-362**, **2019.**

AUTHORS PROFILE

Mr.N.E Maidoh, holdsB.Sc (Computer Science) from the university of Benin (UNIBEN) and M.Sc (Computer Science) from Enugu State University of Science and Technology (ESUT). He works as a lecturer in the department of Computer Science, School of Science, AkanuIbiam Federal Polytechnic, Unwana, Ebonyi State, Nigeria. He is a registered member of Computer Professionals Registration Council of Nigeria (MCPN). He has published in several national and international journals.

