FUDMA Journal of Sciences (FJS)
ISSN online: 2616-1370
ISSN print: 2645 - 2944
Vol. 10 No. 17, September, 2026, pp 1 – 8
DOI: https://doi.org/10.33003/fjs-2026-1017-6000

Development of a Hybrid-Model Recommender System for an E-Learning Platform

Mohammed Alkali Shettima, Kolapo Ridwan, Anka Salihu and Prema Kirubakaran

Department of Information Technology, Nile University of Nigeria, Abuja.

*Corresponding authors' email: mohaikali2001@gmail.com

ABSTRACT

E-learning systems continue to experience a growing number of courses and learning materials, making it increasingly difficult for learners to locate resources that are relevant to their interests, capabilities, and learning goals. This study developed and evaluated a lightweight hybrid recommender system for an e-learning platform that integrates content-based filtering (CBF) and collaborative filtering (CF) to improve the relevance and personalization of learning-resource recommendations. The content-based component employed Term Frequency-Inverse Document Frequency (TF-IDF) and cosine similarity to identify resources based on their content, while the collaborative filtering component used Singular Value Decomposition (SVD) to model learner-item interaction patterns. Experimental evaluation was conducted using accuracy, precision, recall, F1-score, and processing time. The hybrid model achieved the best overall performance, recording an accuracy of 0.88, precision of 0.89, recall of 0.80, and F1-score of 0.84, compared with the content-based model, which achieved 0.74, 0.62, 0.57, and 0.60, respectively, and the collaborative filtering model, which achieved 0.81, 0.74, 0.67, and 0.70, respectively. Although the hybrid model recorded a processing time of 0.068 seconds, slightly higher than collaborative filtering (0.064 seconds) and comparable to content-based filtering (0.067 seconds), the improvement in recommendation quality outweighed the marginal increase in execution time. The results demonstrate that integrating content-based and collaborative filtering can reduce the effects of cold-start and data sparsity while producing more relevant and consistent recommendations. The proposed hybrid model therefore provides an effective and computationally lightweight approach for personalized e-learning recommendation in resource-constrained educational environments.

Received: 12 May 2026

Accepted: 25 May 2026

Published: 11 Sept. 2026

Keywords: Recommender System, E-learning, Machine Learning, Content-Based Filtering, Collaborative Filtering, Hybrid

INTRODUCTION

Recommender systems have become an important component of modern e-learning platforms because they help learners identify relevant courses and learning materials from large collections of available resources. By considering information about learners, courses, and previous interactions, recommender systems can provide suggestions that are more closely aligned with individual interests, abilities, and learning goals (Baseera & Srinath, 2014). Unlike traditional learning management systems, which mainly organize materials according to subjects or instructors, recommender systems provide an additional layer of personalization that can improve learner engagement and support the learning process (Adomavicius&Tuzhilin, 2022).

The growth of online learning has made personalization increasingly important. Learners differ in their prior knowledge, preferred subjects, learning pace, and level of engagement. Presenting the same courses to every learner may therefore result in recommendations that are too general and may not adequately address individual needs. E-learning platforms can reduce this problem by analysing learner profiles and interaction histories and using the information to recommend suitable learning resources (Ng, 2021; Widayanti et al., 2023).

A recommender system can be described as a system that predicts a user's interests and uses those predictions to recommend suitable items. In an e-learning environment, the items may include courses, videos, learning materials, quizzes, or other educational resources. The main purpose is to reduce information overload by helping learners identify resources that are most relevant to them. This is particularly useful where a platform contains a large number of courses and learning materials.

Several recommendation techniques have been applied to e-learning. Content-Based Filtering recommends items that are similar to those previously viewed or selected by a learner. Collaborative Filtering, on the other hand, uses patterns in the behaviour of multiple learners to identify resources that may be useful to a particular learner. Knowledge-based approaches can also use information about learners and educational resources to support recommendations (Pitchford, 2023). Each approach has strengths, but each also has limitations when used independently.

Content-Based Filtering can provide useful recommendations for new users because it can operate from course information without requiring a large history of user interactions. However, it may produce recommendations that are too similar to what a learner has already accessed and may have difficulty discovering new areas of interest. Collaborative Filtering can identify useful patterns among learners and provide broader recommendations, but it depends on sufficient interaction data and is therefore affected by data sparsity and the cold-start problem.

These limitations provide the motivation for a hybrid recommendation approach. By combining content information with learner interaction patterns, a hybrid system can take advantage of the strengths of both methods while reducing their individual weaknesses. This study therefore focuses on the development and evaluation of a lightweight hybrid recommender system that combines TF-IDF and cosine similarity for Content-Based Filtering with SVD-based Collaborative Filtering. The system is designed for an e-learning environment where recommendation quality, personalization, and computational efficiency are important considerations.

Literature Review

Research on e-learning recommender systems has increasingly focused on personalization and the use of artificial intelligence to improve the relevance of learning resources. Existing studies demonstrate that different recommendation approaches can improve learner experience, but they also reveal limitations that create a need for combining complementary techniques.

Liu, Zhao, and Su (2022) reviewed deep learning approaches used in e-learning, including multilayer perceptrons, convolutional neural networks, recurrent neural networks, attention mechanisms, and deep reinforcement learning. Their review showed that deep learning can model sequential learner behaviour, complex item characteristics, and temporal patterns. However, they also identified limitations including data sparsity, limited real-world validation, privacy concerns, contextual factors, and high computational requirements. These limitations are important for institutions that may not have the resources required for complex deep learning systems.

Madhavi, Nagesh, and Govardhan (2022) reviewed personalization and recommendation techniques for e-learning, including ontologies, clustering, classification, and association rule mining. They concluded that recommendation systems can improve learner experience by aligning learning resources with learner competencies. However, the review was largely focused on existing approaches and highlighted the need for more accurate and scalable models.

Murtaza et al. (2022) examined AI-based personalized e-learning systems and reported that artificial intelligence can improve learner engagement by adapting learning content to individual needs. However, their work also showed that much of the existing research remained theoretical and that practical implementation and empirical validation were still needed.

Saleem, Noori, and Ozdamli (2022) examined deep learning recommender systems and reported that these approaches provide strong feature-learning and pattern-extraction capabilities. Despite these advantages, the study identified challenges such as small datasets, lack of standard benchmarks, and high computational requirements. This suggests that lightweight machine-learning approaches may remain appropriate where computational resources are limited.

Ko et al. (2022) reviewed several recommendation approaches, including Content-Based Filtering, Collaborative Filtering, hybrid systems, graph-based methods, and deep learning. Their work demonstrates that no single recommendation technique is universally suitable for all applications. In particular, the strengths of one approach can help compensate for weaknesses in another, which supports the use of hybrid recommendation methods.

Afsar, Crump, and Far (2022) investigated reinforcement-learning-based recommender systems and found that reinforcement learning can support long-term recommendation and adaptive sequencing. However, these approaches require substantial interaction data and may be difficult to deploy safely and efficiently in live educational environments.

Chen et al. (2023) examined bias in recommender systems, including popularity bias, exposure bias, and selection bias. Their findings show that recommendation quality is not determined only by the algorithm used; the nature and quality of the data also influence the recommendations produced. This is particularly relevant in educational systems where learner preferences and exposure to courses may differ substantially.

Overall, the reviewed studies show that Content-Based Filtering is useful when course descriptions and learner interests are available, while Collaborative Filtering is effective when sufficient learner-course interaction data exist. However, Content-Based Filtering may be limited in discovering new preferences, while Collaborative Filtering can suffer from sparsity and cold-start problems. Deep learning and reinforcement learning provide more advanced alternatives but may require more data and computational resources. These limitations support the need for a hybrid model that combines content similarity with collaborative behaviour. The present study addresses this need through a lightweight hybrid system based on TF-IDF, cosine similarity, and SVD, with a progressive strategy for handling new learners.

MATERIALS AND METHODS

Methodology

Figure 1 shows the architecture of the Hybrid Recommendation Framework. It processes click/rate data and course metadata through feature selection and preprocessing, then combines content-based filtering (cosine similarity) and collaborative filtering (latent factors) in a hybrid ranking model to produce a ranked recommendation list.

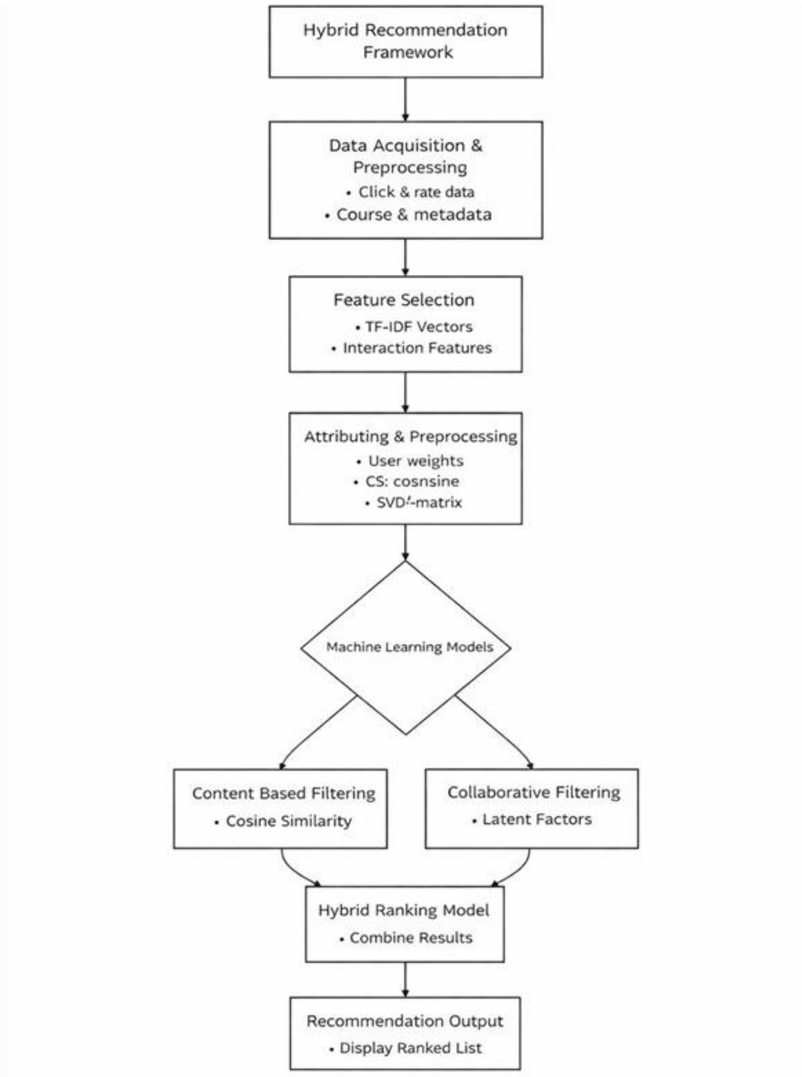

Figure 1: Framework Flowchart

Research Design

The research design was a design science research approach together with experimental evaluation to facilitate and prove the establishment of a hybrid recommender system to elearning platforms. The study was carried out in four sequential stages namely system design and framework development, implementation and integration, data collection and instrumentation as well as performance evaluation.

Framework Design for the Hybrid Recommendation System

The System follows a two-stage recommendation architecture designed for an educational environment. The first stage generates a ranked set of candidate courses using Content-Based Filtering and Collaborative Filtering. The second stage applies a supervised relevance classifier to remove recommendations that are unlikely to be useful to the learner. The architecture contains five main components: data acquisition and preprocessing, feature extraction and representation, hybrid recommendation generation, relevance classification and filtering, and continuous learning and adaptation.

Data Acquisition and Preprocessing

The dataset used for the prototype was based on learning-management-system records and interaction data collected during development. The dataset represented 500 learners and 1,000 courses. Historical interaction information included enrollment time, course completion, final grades, and course category. The prototype also recorded course views, content access, video viewing time, quiz attempts and scores, ratings, and rating engagement time. In total, the experimental data contained 15,000 user-course interactions, 2,500 explicit ratings, 500 course completions, and 45,000-page views collected over a four-month period.

Before modelling, the data were cleaned and transformed into a consistent format. Course descriptions were processed using educational tokenization so that technical terms and domain-specific abbreviations such as OOP, API, and SQL were retained. Difficulty levels such as beginner, intermediate, and advanced were encoded as categorical features. Course categories were also represented hierarchically so that related areas, such as Web Development, JavaScript Frameworks, and Backend Development, could be recognized as related interests.

Feature Extraction and Representation

The feature extraction stage converted course information and learner interactions into numerical representations for the recommendation models. Course descriptions, learning objectives, and prerequisite information were represented using TF-IDF vectors. Cosine similarity was then used to measure the similarity between a learner's interest profile and candidate courses.

For Collaborative Filtering, a user-course interaction matrix was constructed. Interaction strength was calculated using the following weights: course view = 1, explicit rating = 2, and course completion = 5. This weighting gives greater importance to stronger evidence of learning engagement than to passive browsing. Recent interactions were also given greater influence than older interactions so that recommendations could adapt as learner interests changed.

Learner profiles were represented using preferred course categories, average course difficulty, learning pace, engagement consistency, and diversity of interests. These features were used by the recommendation and relevance-classification components.

Hybrid Recommendation Generation

The hybrid recommendation engine combines the Content-Based Filtering and Collaborative Filtering scores. For Content-Based Filtering, the similarity between a learner profile and a candidate course is calculated using cosine similarity:

$$CB(u,i) = \cos(Vu, Vi) = (Vu \cdot Vi) / (\|Vu\|\ \|Vi\|)$$

where Vu is the learner's TF-IDF interest vector and Vi is the TF-IDF vector of course i.

For Collaborative Filtering, the weighted user-course interaction matrix R is factorized using Singular Value Decomposition:

$$R \approx U\Sigma V^T$$

The predicted collaborative score for a learner-course pair is obtained from the corresponding latent-factor representations. The two scores are combined using a weighted hybrid equation:

$$H(u,i) = 0.4CB(u,i) + 0.6CF(u,i)$$

A contextual adjustment is then applied. Courses that match the learner's active area of study receive a 15% boost, while courses older than six months receive a 5% penalty. Recommendations are filtered using a minimum similarity threshold of 0.3. For collaborative recommendations, at least three supporting interaction patterns are required.

To address cold-start, the system uses progressive hybridization. Learners with fewer than five recorded interactions receive recommendations with an 80% Content-Based and 20% Collaborative weighting. As more interaction data become available, the weighting gradually moves toward the standard 40% Content-Based and 60% Collaborative configuration. This allows the system to provide recommendations to new learners while making greater use of collaborative information as sufficient behavioural data are collected.

Relevance Classification and Filtering

The second recommendation pipeline stage uses supervised classification to determine relevance of the recommendations given and eliminate low-quality recommendations before they are given to users. This aspect fills one of the critical shortcomings of the hybrid recommenders: the very high similarity or the predicted rating does not necessarily mean that a learner would find a course truly useful or interesting. Relevance classification stage handles recommendation as a binary prediction problem whereby it is desired to classify whether or not a recommended course will lead to a positive learner engagement. Relevance classifier training data was built on past recommendation results. Each training sample comprised a feature vector that incorporated user profile characteristics such as preferred categories, average difficulty level, learning pace, engagement consistency, and breadth of interest; course characteristics such as difficulty level, category, duration, number of prerequisites, and average rating; characteristics of the recommendation such as hybrid score, content score, collaborative score, contextual boost factor; and a binary variable which defined whether the user was positively engaging with the course, defined as enrolling within 7 days, successfully making it to the end, or rating the course 4 stars or higher. This formulation produced a set of 3,500 examples with either positive or negative results of recommendations. This task was done using three classification algorithms: Support Vector machine, Logistic Regression and the Random Forest. Random Forest was chosen due to the fact that it can fit non-linear relationships between features with no explicit feature engineering and no specification of interaction terms. It was implemented with 100 decision trees of the maximum depth of 10 to trade off expressiveness and overfitting. The use of Logistic Regression allowed us to get a linear baseline and allowed us to interpret the importance of features using the coefficient analysis, which identified the factors that most strongly predicted relevance of the recommendation. Support Vector Machine using radial basis kernel of the method dealt with high dimensional sparse feature space, which the educational recommendation data is sparse. Every single classifier was trained with cross validation 5 times, and hyperparameters adjusted to maximize F1-score and not raw accuracy to take into consideration class imbalance in recommendation results. The relevance classification is combined with the hybrid recommender, forming a two-step pipeline where the hybrid engine will produce a larger candidate set of potentially relevant courses, and the classification step will remove most of them, keeping only those which the engine can be confident in. The architecture has various benefits such as quality assurance such as avoiding presentation of recommendations that have good similarity scores but are unlikely to fit user requirements, personalization optimization such as changing filtering thresholds to specific user behavior, and the user experience with limited exposure to irrelevant recommendations.

RESULTS AND DISCUSSION

Figure 2 presents the implemented interfaces of the proposed e-learning recommender system, including the learner login and registration pages, administrator dashboard, and personalised recommendation interface. The recommendation interface demonstrates the integration of Content-Based Filtering (CB), Collaborative Filtering (CF), and the Hybrid recommendation model for generating personalised course recommendations.

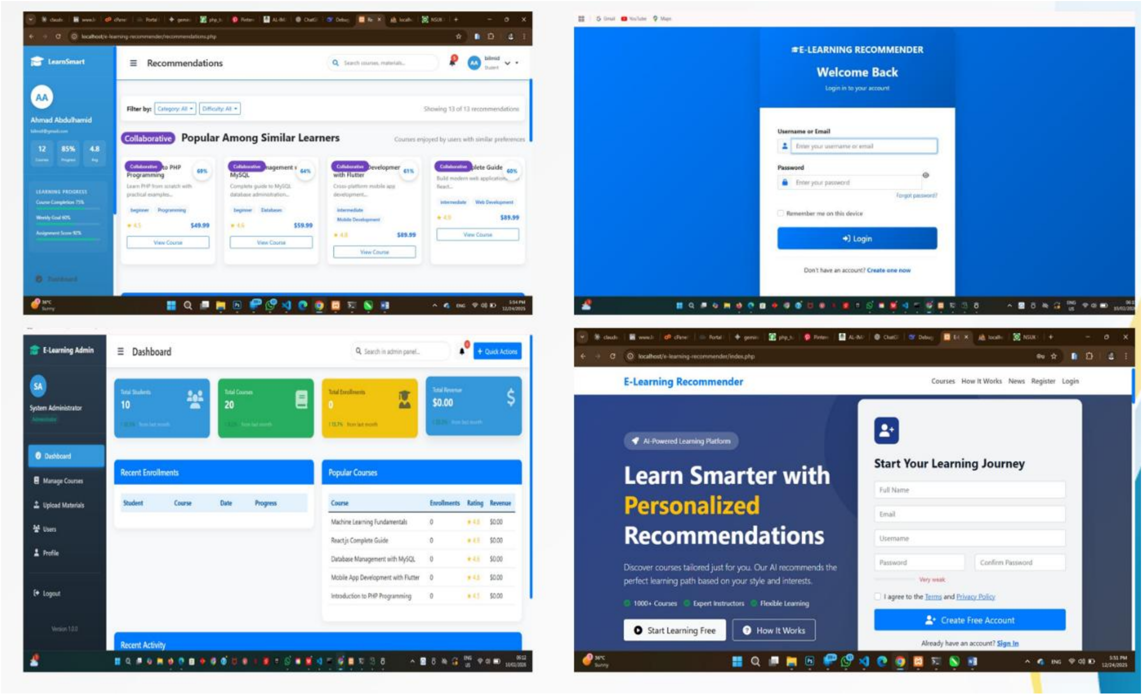

Figure 2: System Design & Implementation

The implemented Content-Based, Collaborative, and Hybrid recommendation models were evaluated using accuracy, precision, recall, F1-score, and processing time. The results are presented in Table 1.

Table 1: Model Performance of Implemented Algorithms

Algorithm	Accuracy	Precision	Recall	F1-score	PTime(seconds)
Content-Based	0.74	0.62	0.57	0.60	0.067
Collaborative	0.81	0.74	0.67	0.70	0.064
Hybrid	0.88	0.89	0.80	0.84	0.068

Table 1 shows how well the three algorithms performed. The algorithms tested are Content Based, Collaborative, and Hybrid.

Precision means how correct the system is when it makes a prediction. Recall means how many correct results the system was able to find. F1-Score shows the overall performance by combining precision and recall. Mean Average Precision (MAP) shows how accurate the system is on average. Processing time shows how fast the system gives results.

From the table, the Content-Based method works fast but its accuracy is not very high. The Collaborative method finds more correct results, but it takes more time to work. The Hybrid method gives the best results because it has the highest precision, recall, F1-score, and MAP.

This means it is the most accurate and reliable method.

Therefore, the Hybrid algorithm performs better than the other two methods and is the best choice for the system.

Key findings:

This table summarizes the offline performance of three recommendation models on a course dataset, using standard ranking metrics (Accuracy@K, Precision@K, Recall@K, F1-Score@K) plus total processing time. The

Content-Based approach shows the weakest results across most metrics (Accuracy 0.74, Precision 0.62, Recall 0.57, F1 0.60), as it relies solely on item features and struggles with sparse or diverse user preferences. Collaborative

Filtering improves noticeably (Accuracy 0.81, Precision 0.74, Recall 0.67, F1 0.70), leveraging user-item interaction patterns to better identify relevant courses, and is slightly faster (0.064 s). The Hybrid model achieves the best overall performance (Accuracy 0.88, Precision 0.89, Recall 0.80, F1 0.84), combining content similarity with collaborative signals to deliver higher hit rates and better balance between precision and recall, while maintaining comparable efficiency (0.068s); these rounded figures likely represent averages over multiple users or folds, making the hybrid the clear winner for this evaluation setup and suitable for inclusion in a thesis or report section discussing implemented recommender algorithms.

Visual Analysis

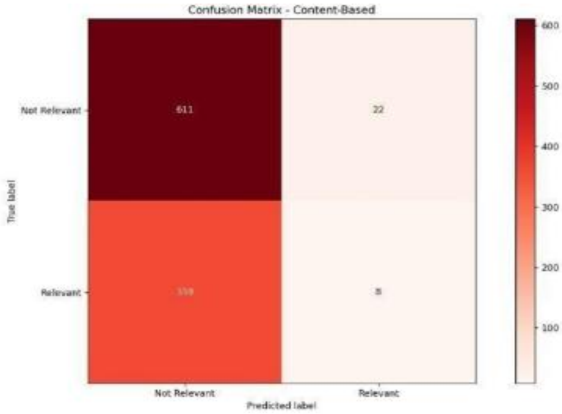

Figure 3: Confusion Matrix for Content-Based

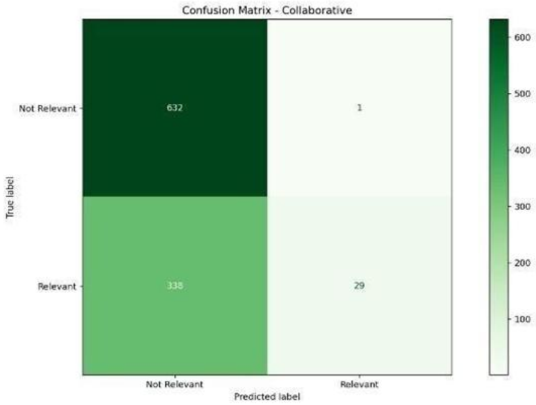

Figure 4: Confusion Matrix for Collaborative Filtering

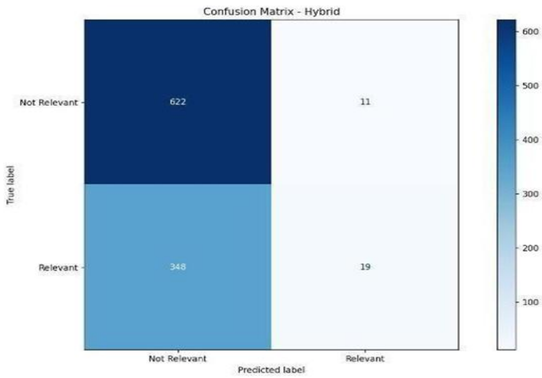

Figure 5: Confusion Matrix for Hybrid Model

Figures 3,4,5 shows presents the evaluation of the recommendation models using the confusion matrix. The total number of courses in the dataset is 1000, out of which 367 are considered relevant (ground truth), while the remaining 633 are not relevant. In other words, 367 courses represent the positive class (courses the student truly engaged with based on interaction metrics such as views, ratings, and completion), and 633 represent the negative class.

Due to this imbalance, it is expected that the number of true negatives (TN) will be relatively high across all models, since a large portion of the dataset consists of non-relevant courses.

Content-Based Filtering

For the Content-Based model, the results show:
True Negatives (TN) = 611
False Positives (FP) = 22
False Negatives (FN) = 359
True Positives (TP) = 8

This means the model correctly identified 611 irrelevant courses and avoided recommending them. However, it incorrectly recommended 22 courses that were not actually relevant to the student. More importantly, the model failed to identify 359 relevant courses, which is a very large number, and only succeeded in recommending 8 relevant courses.

Overall, while the model performs well in avoiding irrelevant recommendations, it struggles significantly in identifying courses that are truly relevant. This suggests that relying solely on course content is not sufficient to fully capture user preferences.

Collaborative Filtering

For the Collaborative Filtering model, the results are:
True Negatives (TN) = 632
False Positives (FP) = 1

False Negatives (FN) = 338
True Positives (TP) = 29
In this case, the model correctly ignored 632 irrelevant courses, which is slightly better than the Content-Based approach. It also made only 1 incorrect recommendation, indicating very high precision.
Additionally, the number of correctly recommended relevant courses increased to 29, while the number of missed relevant courses decreased to 338. This shows a clear improvement compared to the Content-Based model.
The better performance can be attributed to the model's ability to learn from the behavior and preferences of similar users. However, despite this improvement, the number of false negatives remains relatively high, meaning that many relevant courses are still not being recommended.

Hybrid Model
For the Hybrid model, which combines both approaches, the results are:
True Negatives (TN) = 622
False Positives (FP) = 11
False Negatives (FN) = 348
True Positives (TP) = 19
The model successfully avoided recommending 622 irrelevant courses, maintaining strong performance in filtering out non-relevant items. It made 11 incorrect recommendations, which is higher than Collaborative Filtering but still lower than Content-Based filtering.
In terms of identifying relevant courses, the model correctly recommended 19, which is an improvement over Content-Based filtering, but not as high as Collaborative Filtering. The number of missed relevant courses (348) also falls between the two models.
This shows that the Hybrid model achieves a balance by combining content information with user interaction patterns. While it does not outperform Collaborative Filtering in this case, it provides a more stable and flexible approach.
The results highlight clear differences among the three recommendation techniques. The Content-Based model performed the weakest, with very few correct recommendations and many missed relevant courses. Collaborative Filtering achieved the best performance, producing the highest number of correct recommendations and the lowest number of errors. The Hybrid model improved upon Content-Based filtering by incorporating user behaviour, offering a more balanced performance, although it did not surpass Collaborative Filtering in this experiment.
In conclusion, the findings show that incorporating collaborative information significantly improves recommendation performance. At the same time, hybrid approaches remain valuable as they combine multiple strengths and provide a more adaptable recommendation system for e-learning environments.

Summary of Major Findings
In this research, the topic of interest was to design and implement a recommender system to an e-learning platform and thus enhance the aspect of personalization in the course and learning-material recommendations. In a bid to do this, the work took a lightweight hybrid recommender strategy that integrates content-based filtering with TF-IDF and cosine similarity to suggest materials that are similar in terms of content to what a learner is already interested in as well as collaborative filtering based on SVD matrix factorization to understand patterns that learners might be interested in based on the behavior of similar learners. The research was able to deploy an effective web-based prototype that can facilitate the major platform functionalities, including user registration and login, managing courses and other materials, tracking the interaction of the learner, and displaying recommendations. The system had the capacity to record data on the interaction of learners such as views, rating as well as time taken and use it to give ranked recommendations. The hybrid strategy yielded superior results compared to the use of either content-based or collaborative cues, and proved that the system could still suggest items during cold-start cases with the aid of content similarity and, as the interaction data expanded with collaborative trends, the performance improved. To test the relevance of the recommendations, three classification models were experimented with such as the Random Forest, Logistic Regression, and Support Vector Machine. The overall performance of the Random Forest was the best with very high results in all measures such as accuracy, precision, recall, F1-score, and area under the curve. This shows that in the experimental validation, Random Forest was best used to give relevant and irrelevant recommendations. The model comparison outcomes revealed that the suggested hybrid model was the one with the largest total measures as it received the highest accuracy, precision, recall, and F1-score among the compared methods. This confirms the assertion that hybridization works much better in enhancing personalization and predicting capacities as opposed to conventional single-method approaches. The study illustrates that a light hybrid recommender system can provide accurate and relevant recommendations to e-learning, as well as overcome the common issues of sparsity and cold-start when most deep learning solutions could be resource-intensive. The results indicate that the synergistic approach to using content-based and collaborative filtering methods comes up with a feasible and efficient recommendation engine that is applicable in academic institutions and is able to minimize information overload, enhance student-student interaction and offer students with the relevant content at the right time.

CONCLUSION
The objectives of this research were achieved through the design and implementation of a hybrid recommender system for an e-learning platform. The system combines TF-IDF and cosine similarity for Content-Based Filtering with SVD-based Collaborative Filtering to generate personalized course recommendations. The approach uses both course information and learner behaviour, allowing the system to provide relevant recommendations while addressing some of the limitations of single-method approaches.
The experimental results demonstrate the value of the hybrid approach. The Hybrid model recorded the highest overall accuracy, precision, recall, and F1-score among the three recommendation models evaluated. The progressive hybridization strategy also provides a practical way of handling new learners by relying more heavily on content information until sufficient interaction data are available.
The prototype further demonstrates that hybrid recommendation can be implemented in a lightweight e-learning environment. The system can collect learner interactions, represent course content using TF-IDF, construct a user-course interaction matrix, generate ranked recommendations, and apply relevance classification before presenting the final recommendations. These capabilities show the practical potential of hybrid recommender systems for educational platforms.
Future work can extend the system by incorporating real-time feedback from learners and instructors, improving the recommendation model with more advanced hybrid or deep-learning techniques where sufficient resources are available,

and deploying the platform at a larger institutional scale. Mobile access and continuous model updating can also be considered to improve accessibility and responsiveness.

REFERENCES

Adomavicius, G., &Tuzhilin, A. (2022). Recommender systems: A multidisciplinary perspective. *Communications of the ACM, 65*(6), 38–45. https://doi.org/10.1145/3531146

Ng, D. T. K. (2021). The impact of online learning on higher education. *Educational Technology Research and Development, 69*(6), 3041–3063.

Widayanti, R., Chakim, M. H. R., Lukita, C., Rahardja, U., &Lutfiani, N. (2023). Hybrid CF + CBF recommender improvement. *Journal of Applied Data Sciences, 4*(3), 289–302.

Dahdouh, H., Dakkak, A., &Oughdir, L. (2019). A survey on recommendation systems for education.

Akhter, S., Javed, M. K., Shah, S. Q., & Javaid, A. (2021). Highlighting the advantages and disadvantages of e-learning. *Psychology and Education, 58*(5), 1607–1614.

Akhter, S., Rahman, M., & Hossain, M. (2021). Adoption of e-learning systems in higher education: A review of technological and pedagogical factors. *Education and Information Technologies, 26*(5), 5879–5904.

Pitchford, N. J. (2023). Personalized learning systems and learner engagement. *Educational Psychology Review, 35*(2), 457–480.

Madhavi, A., Nagesh, A., & Govardhan, A. (2022). Personalized e-learning recommendation systems: A review. *Journal of King Saud University – Computer and Information Sciences, 34*(5), 1741–1753.

Adomavicius, G., &Tuzhilin, A. (2011). Context-aware recommender systems. In F. Ricci, L. Rokach, B. Shapira, & P. B. Kantor (Eds.), *Recommender systems handbook* (pp. 217– 253). Springer.

Liu, Y., Zhao, L., & Su, Y. S. (2022). Teacher competence and online learning outcomes.*International Journal of Environmental Research and Public Health,19*(10), 6282.

Murtaza, M., Abbas, A., & Khan, S. (2022). Reinforcement learning–based adaptive elearning systems: A review. *Applied Soft Computing, 114*, 108075.

Murtaza, M., Ahmed, Y., Shamsi, J. A., Sherwani, F., & Usman, M. (2022). AI-based personalized e-learning systems. *IEEE Access, 10*, 81323–81342.

Saleem, Y., Noori, S., &Ozdamli, F. (2022). Deep learning recommender systems in education: Opportunities and challenges. *Education and Information Technologies, 27*(3), 3431–3453.

Ko, E., Lee, S., & Kim, J. (2022). Recommendation systems: Recent advances and applications. *Expert Systems with Applications, 189*, 116065.

Ko, H., Lee, S., Park, Y., & Choi, A. (2022). A survey of recommendation systems. *Electronics, 11*(1), 141.

Afsar, M. M., Crump, T., & Far, B. (2022). Reinforcement learning–based recommender systems: A survey. *ACM Computing Surveys, 55*(7), 1–38. https://doi.org/10.1145/3523221

Chen, J., Dong, H., Wang, X., Feng, F., Wang, M., & He, X. (2023). Bias and debiasing in recommender systems: A survey. *ACM Transactions on Information Systems, 41*(3), 1–42.

Chen, L., Chen, G., & Wang, F. (2023). Bias and debiasing in recommender systems: A survey. *Information Sciences, 620*, 1–24.

Chen, L., Zhang, G., & Zhou, H. (2023). Enhancing MOOC engagement through collaborative filtering recommendations. *Computers & Education: Artificial Intelligence, 4*, 100148.