



PREDICTIVE COURSE PLACEMENT: UTILIZING AN ARTIFICIAL INTELLIGENCE-BASED APPROACH FOR RECOMMENDATIONS THROUGH THE APPLICATION OF COSINE SIMILARITY AND MULTI-MODEL ARCHITECTURES

Manocheer C. Alipour*¹

¹Bachelor of Science in Information Technology Program, Isabela State University, Angadanan, 3307, Isabela, Philippines.

¹<https://orcid.org/0009-0006-9594-8233>

Email: *Manocheer.c.alipour@isu.edu.ph

ARTICLE INFO

Article History

Received: February 23, 2026

Reviewed: June 10, 2026

Accepted: July 10, 2026

Published: July 31, 2026

Keywords:

Recommendation model,
Educational Data Mining,
Machine Learning,
Ensemble Model,
XGBoost.

ABSTRACT

Accurately placing students in appropriate academic programs, such as determining whether they are better suited for a particular course, is fundamental to student success and retention. This study aims to construct a recommendation model that uses data mining and machine learning techniques to guide currently enrolled students in their program to make better selections. The research proposes a multi-model architecture that integrates a content-based cosine similarity matching component with three distinct machine learning classifiers, namely Logistic Regression, Random Forest, and XGBoost. In addition, the models were trained and evaluated using 10-fold cross-validation. Performance evaluations revealed that XGBoost was the most effective individual classifier, achieving a mean accuracy of 91.7%, an ROC-AUC of 97.5%, and a positive-class F1 score of 90.4%. The ensemble model also demonstrated strong predictive and recommendation performance, achieving 90.6% accuracy and 96.8% ROC-AUC. Logistic Regression achieved an accuracy of 88.7%, while the Random Forest model lagged with an accuracy of 83.8%. The findings demonstrate that XGBoost and Ensemble models are the most reliable and effective classifiers for predicting student course placement on this specific dataset. This hybrid recommendation engine successfully translates historical and input data into actionable, data-driven insights to help students make informed decisions about their academic careers.



Copyright ©2026 by authors and Galileo Institute of Technology and Education of the Amazon (ITEGAM). This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

I. INTRODUCTION

The fast and unpredictable life demands a recommendation engine tool that recommends content to users and machines based on their preferences and interests, using algorithms to analyze their needs, likes, and wants, drawing on historical data, user experience, academic performance, business interests, and other relevant factors, to create individualized recommendations [1-3]. Ensemble recommendation models [4], [5] in education combine multiple sources of information or numerous recommendation strategies to improve recommendations. The society, which relies heavily on knowledge and information interaction, is a key component in technological innovation and a fundamental tool for developing teaching methodologies. Many policymakers in developing countries consider upper-secondary technical and vocational education and training critical to economic growth and poverty reduction [6], [7]. The education system in the Philippines aims to develop high-level professionals who directly impact leadership, business, and employability, advance knowledge through research, and apply new knowledge to improve the quality of human life, industrialization, development, and careers [8]. These recommendation models may include collaborative filtering, content-based filtering, matrix factorization, and hybrid approaches that integrate multiple techniques to improve recommendation accuracy [9-13]. Cosine similarity [14-16] is a metric used in data analysis to evaluate the degree of similarity that is not zero and located within an inner product space between two vectors. This angle is determined by dividing the dot product of the vectors by the product of their magnitudes. The cosine similarity is not dependent on the magnitudes of the vectors, which are solely governed by the angle by which they are oriented.

It is always the case that the cosine similarity falls inside the range of $[-1, 1]$. The knowledge and understanding to classify the student's academic courses/track have been observed and challenged for a long time. It is regarded as an essential topic of study across a variety of fields, as it benefits both families, students, and decision-makers. However, the primary purpose of the research was to construct a model that can be recommended to students in the field of information technology using recommendation techniques. This pertains to the research study that used a recommender model and data mining techniques to provide results on student academic performance. Accurately placing students in appropriate courses or academic programs is fundamental to student success and retention. Course placement in this context refers to assigning students to the most suitable academic program or track, for example, determining whether a student successfully graduates from the Bachelor of Science in Information Technology (BSIT) program or the Bachelor of Science in Computer Science (BSCS) program based on their prior and current achievements, skills, and characteristics. Proper placement ensures that students are neither overwhelmed nor under-challenged, thereby improving learning outcomes and satisfaction. Traditionally, such placement decisions have relied on entrance exams, counselor judgments, and heuristics that may not fully capture a student's multifaceted background. The increasing availability of educational data has opened the door for data-driven approaches to augment these decisions. In recent years, different techniques in Machine Learning (ML) and artificial intelligence (AI) [17-20] been actively explored as tools for predicting student performance and recommending personalized academic paths, not only for students but also for policymakers and decision-makers. These models can analyze huge student datasets to recognize complex patterns and predict academic outcomes. Deep learning models, in particular, can capture nonlinear relations and connections that traditional statistical methods might overlook. For instance, predictive models have been one way used to identify at-risk students in need of intervention, sometimes integrating behavioral data, such as online learning activities, with demographic and grade information to trigger timely support. Applying these techniques to course placement, recommending the most appropriate course or major for a student, poses unique challenges, such as classification problems that include multiple classes (different programs) and require a balance between accuracy, interpretability, and fairness in an academic context. This study addresses the issue of identifying students based on first and second-year subjects, demographics, family background, and other factors to determine whether they should continue the course, since there are a number of students who are studying in different courses however, their grades are more suitable for other courses. The rest of this study is as follows: Section 2, Literature review, describes other researchers' views on the different recommended approaches; Section 3, Methodology, describes in detail how the ensemble approach works, from data input to final predictions. Furthermore, Section 4 Results and discussions present evaluations of metrics, and results from distinct algorithms and an ensemble pipeline using various metrics. Lastly, Section 5 concludes by brief the key findings of the study.

II. RELATED WORKS

In this section, researchers explore the different methods and approaches used by other researchers in this field. Logistic Regression (LR) has often been used as a baseline for binary classification tasks, such as predicting dropout or pass/fail, due to its simplicity and interpretability [21-25]. Traditional classification models such as decision trees, logistic regression, and Naïve Bayes have been widely used to predict student success and choices the said algorithms widely used in different systems and approaches, due to their simplicity and interpretability. Complex ensemble methods and hybrid systems have recently gained attention for their superior accuracy. Learning approaches that combine multiple algorithms have been shown to improve robustness and predictive performance across different categories, including educational tasks [26-32]. Combining several classifiers and algorithms, such as SVM, KNN, and linear regression, Random Forest (RF) [33], [34], XGBoost (XGB) [3], [35-37] (but not limited to), results in building an ensemble model this approach in may cases improve accuracy compared to individual models when predicting different categories of data, including student education placements. Deep learning models, especially recurrent neural networks like LSTMs, have also been applied to educational data, where temporal or sequential Patterns are relevant [38]. Researchers found that a LSTM network analyzing students' time-series behavior in a learning management system outperformed eight conventional classifiers in predicting early course performance. The LSTM-based method improved accuracy and earlier detection of at-risk students using clickstream data, highlighting the value of both ensemble methods and LSTMs for educational predictions [39-43]. Content-based methods computing similarities to make recommendations: each student's profile and each course/program's profile are represented as vectors, and the closest match is found. Techniques such as term frequency-inverse document frequency (TF-IDF) [44-46], vectorization, and cosine similarity have been employed in course recommendation systems to measure how well a student's profile matches a course's content or requirements. Use of TF-IDF on course descriptions and cosine similarity to recommend courses to students based on content alignment [47], [11].

II.1 MODEL SELECTION AND EVALUATION

II.1.1 Algorithms

The researcher established a comprehensive and multiple evaluation benchmark for the algorithms used in this study: multinomial Logistic Regression (LR), Random Forest (RF), XGBoost (XGB), and term frequency-inverse document frequency (TF-IDF) by selecting the following algorithms and ensemble models.

II.1.2 Term Frequency-Inverse Document Frequency

TF-IDF The similarity score ranges from -1 to 1 for TD-IDF values. A value of 1 indicates that the vector points in the same direction and is perfectly aligned at 0 degrees. Value 0 indicates that the vectors are orthogonal (perpendicular, with an angle of 90°), i.e., uncorrelated, and a value of -1 indicates that the vectors are pointing in opposite directions (angle of 180°). The system computes cosine similarities for all training documents, identifies the nearest neighbor based on the highest cosine similarity in dataset, and predicts the label for the input content that is most similar to the training data, resulting in the items with the greatest cosine similarities being recommended as the most related.

Term frequency where $tf(t, d)$ stands for Term Frequency: the frequency of a term t in document d . $idf(t)$ stands for Inverse Document Frequency, which measures how much information the word provides. The rarer the word in a document, the higher the value of the word in terms of information.

$$tf(t, d) = \frac{f(t, d)}{\sum_{t'} f(t', d)} \quad (1)$$

Cosine similarity dot product represented by numerator x, v , which shows the inner product of two vectors to represent the product magnitudes (Euclidean, norms, lengths) of vectors x and v represented by $\|x\| \|v\|$.

$$\text{CosineSim sim}_{cos}(x, v) = \frac{x \cdot v}{\|x\| \|v\|} \quad (2)$$

II.1.3 Logistic Regression

Logistic regression directly computes probabilities using the SoftMax function. Many researchers use it for text classification because it provides the probability that each feature has on the task. Softmax regression, or multinomial logistic regression, uses TF-IDF feature vectors to identify and predict content labels. The TF-IDF and logistic regression model compute a score for each possible class via a linear combination. Logit represents the raw, unnormalized score produced before it is passed to an activation function like Softmax. Where the value z_k is referred to as a linear predictor or logit

$$z_k = w_k^T + b_k \quad (3)$$

The SoftMax function gets the z_k to produce a probability distribution over the classes. Where K is the total number of classes. In General Machine Learning Context, linear combination is a fundamental building block across many models.

$$P(y = k|x) = \frac{\exp^{z_k}}{\sum_{j=1}^K \exp^{z_j}} \quad (4)$$

II.1.4 Random Forest

The random forest using the TF-IDF features as input, each decision tree in the random forest produces its own predicted class for the given input, and then aggregates these individual predictions to make a final decision based on majority voting. Nonlinear classifiers in the RF branch can capture interactions among TF-IDF features. Where where $\hat{y}$ is the final predicted value and b is the total number of trees in the forest and $\hat{y}_b(x)$ is the prediction of the b -th decision tree for the input x .

$$\hat{y}_{RF} = \text{mode}(\{\hat{y}_1(x), \hat{y}_2(x), \dots, \hat{y}_b(x)\}) \quad (5)$$

II.1.5 XGBoost

The objective function ($\mathcal{L}$), which includes a loss function term $l(y_i, \hat{y}_i)$ and a regulatory term $\Omega(f_t)$, Regularization helps mitigate the risk of overfitting

$$\mathcal{L} = \sum_i l(y_i, \hat{y}_i) + \sum_i \Omega(f_t) \quad (6)$$

regularized objective where w are the output scores for the leaves with respect to the tree and T is the number of leaves in the given tree; the value of γ controls the minimum loss-reduction gain needed to split an internal node. Shrinkage is an additional regularization parameter in XGBoost that reduces the step size in the additive expansion. A secondary advantage of reduced tree complexity is that models are trained faster and require less storage space. Finally, the complexity of the trees can also be limited using other strategies, such as limiting the tree depth.

$$\Omega(f_t) = \gamma T + \frac{1}{2} \lambda \|w\|^2 \quad (7)$$

II.2 PERFORMANCE METRICS

In this section, the researcher discusses the metrics used to validate the results of this study, applying multiple validation metrics that are widely used by researchers.

Accuracy: as the total number of correctly classified instances and It treats all classes equally. where N is the total number of predictions. , $\hat{y}_i$ is the predicted value for the i -th instance, y_i is the actual true value for the i -th instance and $1(\hat{y}_i = y_i)$ It takes the value 1 for accurate prediction and 0 otherwise, due to the indicator function's binary nature. The $\sum_{i=1}^N$ sums the total number of correct predictions.

$$\text{Accuracy} = \frac{1}{N} \sum_{i=1}^N 1(\hat{y}_i = y_i) \quad (8)$$

Precision Calculation: the precision of a specific class, where k is the ratio of true positives (TP_k) to the total number of instances predicted as positive for that class ($TP_k + FP_k$), includes both true positives and false positives (FP_k).

$$\text{Precision}_k = \left(\frac{TP_k}{TP_k + FP_k} \right) \quad (9)$$

Macro recall measures the classifier's ability to correctly identify instances of each class, averaged equally across classes. It is especially useful when missing minority classes are costly. TP_k : True Positives for class k , the number of instances correctly identified as belonging to that class. FN_k : False Negatives for class k refer to instances that belonged to the class k but were missed by the model. This metric measures the classifier's completeness or sensitivity in identifying all relevant cases of that class.

$$Recall_k = \frac{TP_k}{TP_k + FN_k} \quad (10)$$

Macro F1 is the unweighted mean of per-class F1 scores. It balances precision and recall per class and is robust to class imbalance by preventing dominant classes from overwhelming the metric. The F1 score for class k is the mean of precision and recall, balancing these metrics into a single value. Precision ($Precision_k$) measures the accuracy of positive predictions for class k , while recall ($Recall_k$) identifies actual positive instances. The F1 score is mostly useful for imbalanced datasets as it better reflects model performance on the positive class than simple accuracy.

$$F1_k = \frac{2 \cdot Precision_k \cdot Recall_k}{Precision_k + Recall_k} \quad (11)$$

Area Under the Curve: measures a model's ability to differentiate between positive and negative classes for an individual class k , denoted as (AUC_k). It is an integral of the True Positive Rate (TPR) over the False Positive Rate ($x = FPR$). This integral formulation is a known method for computing the AUC from the ROC curve definition.

$$AUC_k = \int_0^1 TPR(x) dx \quad (12)$$

III. METHODOLOGY

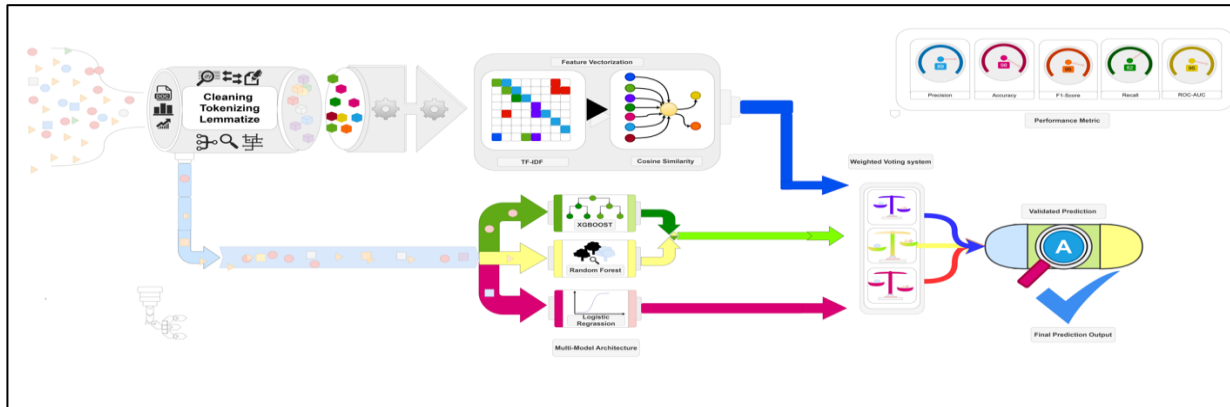


Figure 1: Phase I, II, III illustration.
Source: Author, (2026).

In this section, the researchers explain how the model uses multiple algorithms and processes to select the best direction for students, giving them more chances to complete the course. The process is divided into three Phases: Phase I Input, Phase II Process, and Phase III Result. Figure 1 includes several well-known algorithms, such as multinomial Logistic Regression (LR), Random Forest (RF), XGBoost (XGB), and term frequency–inverse document frequency (TF-IDF), combined into a single ensemble model.

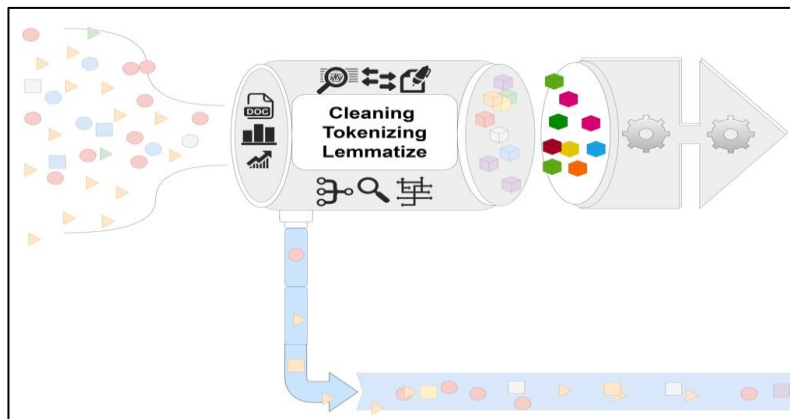


Figure 2: Phase I: Input Data and Preprocessing.
Source: Author, (2026).

III.1 PHASE ONE

Located at the top of the model, is called the input phase shown in Figure 2. It includes input data, data preprocessing and cleaning, and Concatenation and Label Encoding, starting with a dataset that may contain numerical, rating, timestamp, or string data. In the second stage of the first phase, Tokenization & Lowercasing, Noise Removal, stop word Filtering, Label Encoding, and Dataset Splitting, in addition to converting the data into a format for analysis. To address limitations in dataset availability due to data privacy laws, the researchers created the BSIT and BSCS datasets, augmenting them with a Generative Adversarial Networks(GAN) to used with the original dataset and splitting the data into a stratified 80/20 train-test split. The three machine learning models used in this study are Logistic Regression, Random Forest, and XGBoost algorithms. In addition, the 10-fold cross-validation was chosen by the researcher; furthermore, all code and analysis were run in Google Colab using standard virtual machine resources. The dataset attributes that were used in this study are shown in Table 1. After finishing Phase I, all the processed data feeds are sent to Phase II for modeling.

Table 1: Dataset Attributes.

Attribute	Attribute	Attribute
Age	Free time after school	Many hours playing games in week
Sex	Going out with friends	Weekly study time
Food habit	Schoolday alcohol consumption	Easy access to computer
Family size	Weekend alcohol consumption	Easy access to internet
Graduated_from(highschool)	Having_mobile	Own computer
Father educational	Having smartphone	Using a internet per week
Mother_educational	Grade inalgorithms	Grade in introduction of computing
Mother occupation	Number subject failures	Grade in computer programming 1
Father occupation	Grade in physics	Grade in computer programming 2
Your occupation	Grade_in_chemistry	Grade in discrete mathematics
Area flooded during a typhoon or heavy rain	Home to school travel time	Grade in data structures and algorithms
Mode of transportation to school	Romantic_relationship	Grade in quantitative methods
Reason to choose this school	Family relationships	

Source: Author, (2026).

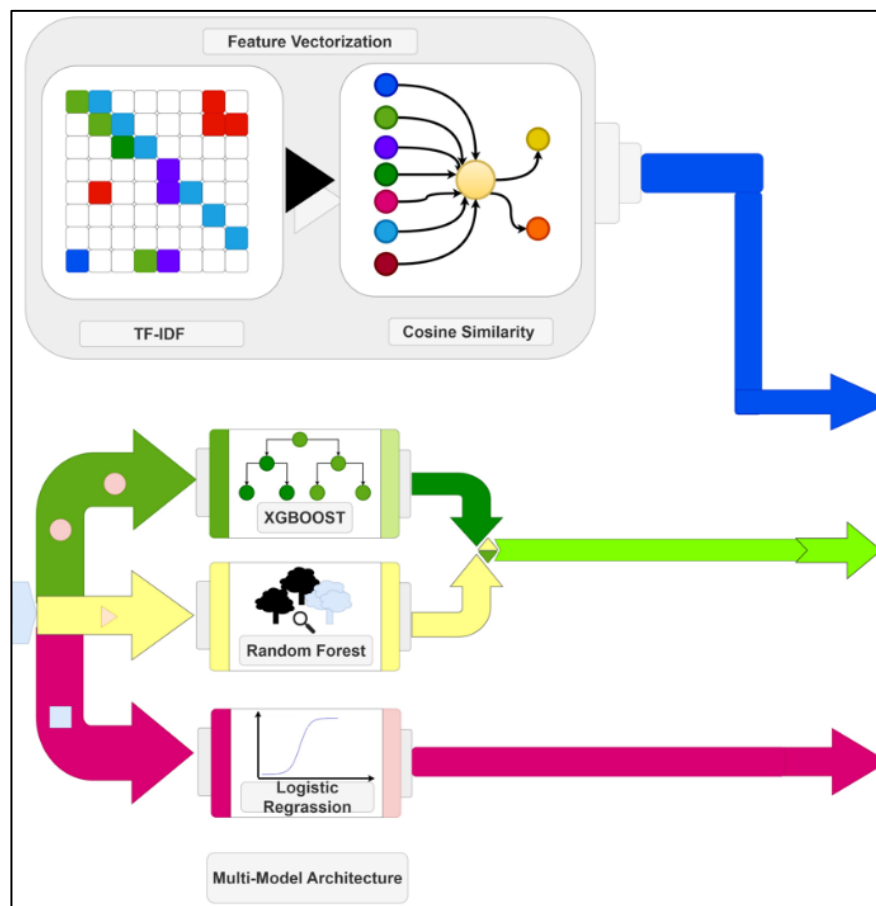


Figure 3: Phase II combined multi-models.

Source: Author, (2026).

III.2 PHASE TWO

Phase two has four multi-models shown in Figure 3: one Similarity-Based Model, two Logistic Regression models, three Random Forest models, and four XGBoost models. The research discusses each in the following:

III.2.1 Logistic Regression

Figure 3 shown the process starts by converting text into a numerical vector for the machine learning algorithms to process in a TF-IDF vectorizer, to compute the weighted sum of each feature, producing one real-valued score (a logit) per class using the SoftMax. The logit indicates the likelihood of each class, and the SoftMax stage converts the logits into a probability distribution over classes. Finally, the algorithm uses these probabilities to output a single class decision.

III.2.2 Random Forest

A technique such as a random forest, an ensemble learning model that uses multiple decision tree methods to find patterns in data. To extract meaningful knowledge from the data, each tree is trained on different subsets of data structures. gives it an advantage in applying structure to provided data and in finding patterns in large-scale datasets, which allows the diversity among the trees to capture a range of patterns, reducing the likelihood of alignment errors. To generate a final prediction, the model typically averages the probability estimates from each individual tree, resulting in a more robust and accurate output. Random Forest learns nonlinear decision rules by combining many diverse trees, the probabilities stage provides calibrated or quasi-calibrated class likelihoods derived from ensemble averaging, and the prediction stage maps those likelihoods to final labels used for reporting results or deployment.

III.2.3 XGBoost

This stage starts with a tokenizer that feeds data into the boosting block. XGBoost is a boosting-style ensemble learning process in which a strong classifier is created by combining a sequence of weak decision-tree classifiers. The process starts with the first box, representing the training samples in their initial form. The process makes an initial prediction, measures the distance to the true labels, and updates the model state. The residuals are passed to the next step, emphasizing that the errors serve as the learning signal for that step. The process repeats this cycle across multiple columns of a weak classifier, indicating that a weak learner (a shallow tree) is fit to the current error structure in the data. The dataset is reweighted or re-focused after each weak learner, so that samples that were previously mis-predicted receive greater influence in the next fitting step.

III.2.4 TF-IDF

TF-IDF vectorizer transforms text into a numerical vector by assigning each term a weight based on how often it appears in a document relative to its frequency across the corpus. Common words receive lower weights, and rare and informative words receive higher weights. The similarity between two vectors is measured as the cosine of the angle between them, enabling comparison of the new input vector with the training data using cosine similarity. After adding each weak learner, the ensemble's predictions change, new residuals are computed, and the training distribution or effective loss landscape is updated for the next learner. By combining multiple weak decisions, the final component of the boosted ensemble model represents the boosted ensemble formed by combining all weak learners, typically as a weighted sum or an additive model, so that each new tree contributes an incremental correction to what has already been learned.

III.3 PHASE THREE

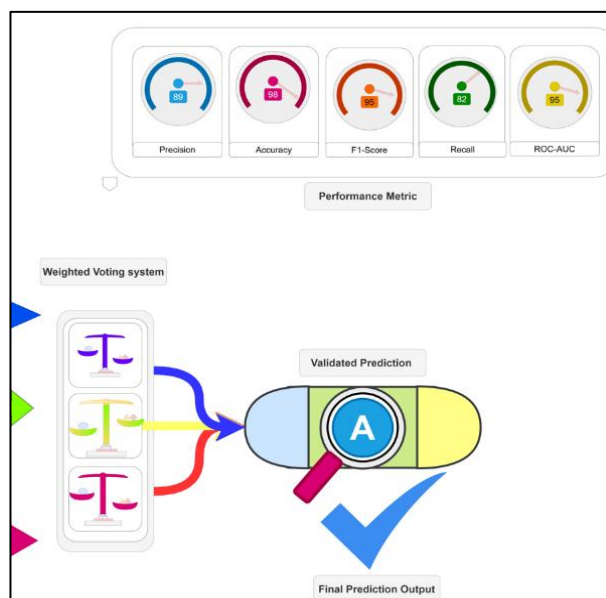


Figure 4: Phase III evaluation.
Source: Author, (2026).

Figure 4 shown by capitalizing on model diversity and focusing on an AI/ML pipeline that ensures the classifications and recommendations are multi-model, and the prediction is accurate and robust. After all models predict the output, all predictions combine to create final predictions for accomplishing this step the researcher used Weighted Voting, different validations technique used to allow the model to decide what is the best perdition for given input data, validations approaches such as accuracy, precision, recall, F1-score, ROC-AU and others validations which offers the most accurate model, more influence in decision making and predictions output.

IV. RESULTS AND DISCUSSIONS

This section discusses the performance of each model used in this study. Table 2 shows the model the mean of from fold (1) up to fold(10), Accuracy Mean, Precision Positive Mean, Recall Positive Mean, F1 Positive Mean, and ROC AUC of the different models used in this study top performers among these combined models are the Ensemble model and the XGBoost, both of which can capture complex, nonlinear patterns and relationships within the data [48-52]. However, this does not mean that the Random Forest and logistic regression underperformed because of the nature of the data provided to the models rather, it shows that different models perform differently across datasets [53]. The researcher evaluates Logistic Regression, Random Forest, XGBoost, and an ensemble. The results for each model are summarized in Table 2, which shows first that Logistic Regression achieves 88.86% accuracy and F1 of 87.43%, and second that the Random Forest classifier performs, attaining 83.86% accuracy and F1-score of 81.71%, third XGBoost scored accuracy of 91.73% and F1 score of 90.46%, the ensemble of all three models also achieved 90.64% accuracy, with an F1-score of 89.19%. The results show that the model can identify the pattern in the dataset and provide a proper recommendation as output.

Table 2: Performance of Classification Models on fold(1) to fold(10) Prediction.

Model	Accuracy Mean	Precision Mean	Recall Mean	F1 Mean	ROC Mean
XGBoost (TFIDF)	0.9173	0.9274	0.8929	0.9046	0.9753
Logistic Regression (TFIDF)	0.8874	0.8800	0.8769	0.8743	0.9693
Ensemble (Logistic Regression (TFIDF)+Random Forest (TFIDF)+XGBoost (TFIDF))	0.9064	0.9086	0.885897436	0.8919	0.9687
Random Forest (TFIDF)	0.8386	0.8410	0.803205128	0.8171	0.9093

Source: Author, (2026).

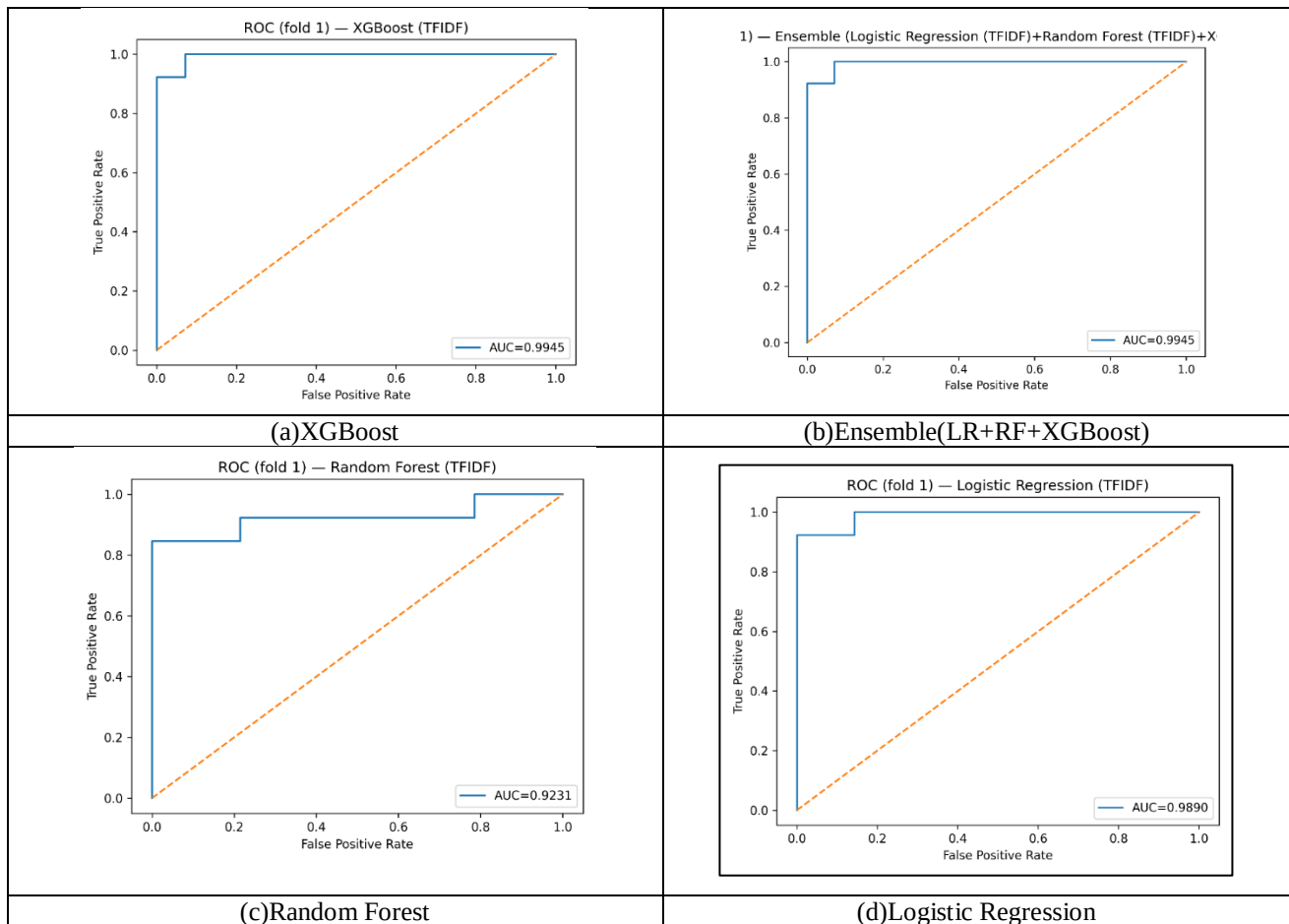


Figure 5: ROC output for fold (1) of XGBoost, LR, RF, and Ensemble model.

Source: Author, (2026).

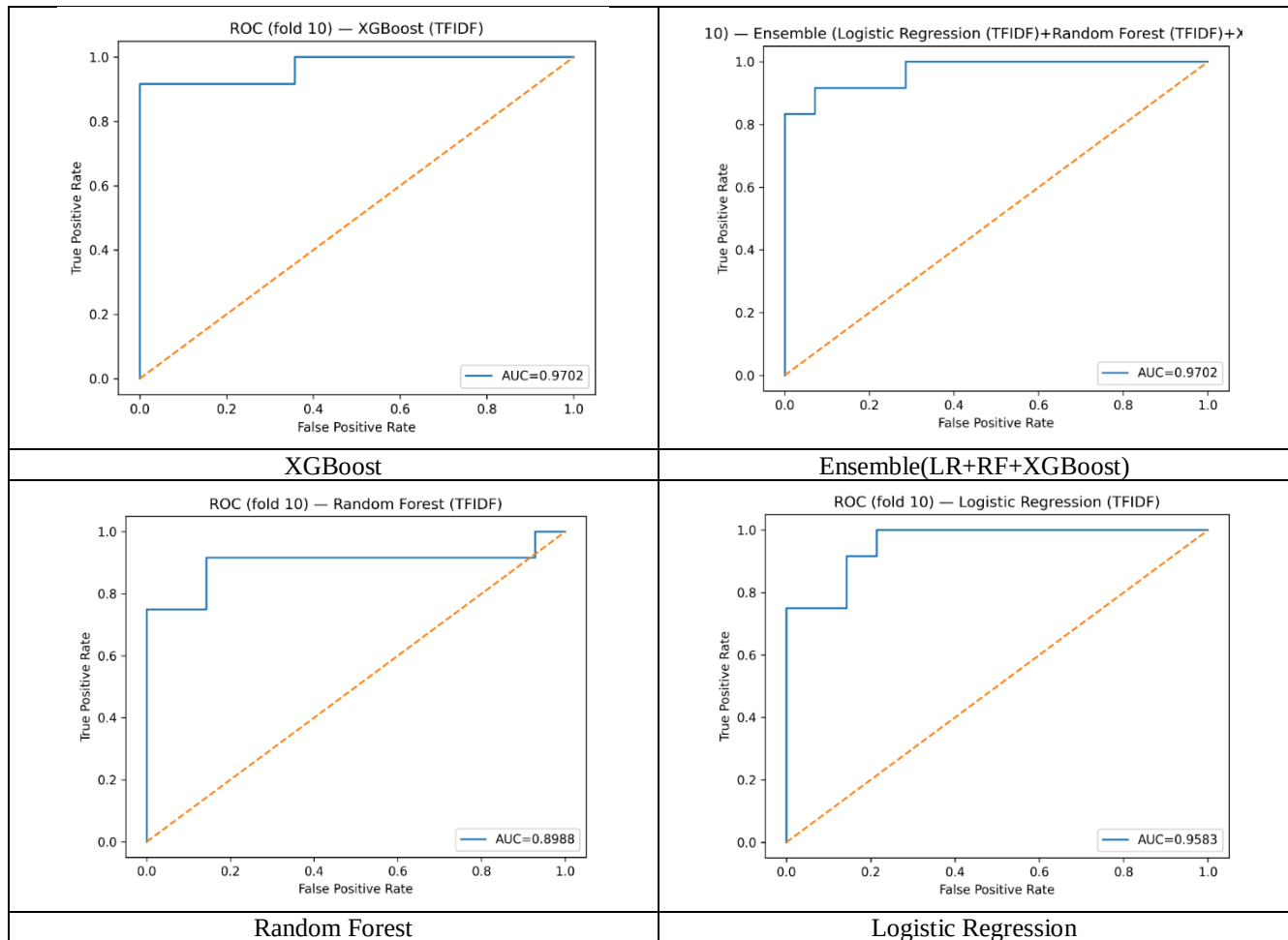


Figure 6: ROC output for fold (10) of XGBoost, LR, RF, and Ensemble model.
Source: Author, (2026).

Figure 5 and 6 show ROC curves derived from fold (1) and fold (10) other cross validation fold is not included, only across all folds for Logistic Regression, Random Forest, XGBoost, and the XGBoost mean score for ROC-AUC is 97.53% and the ensemble model, which achieved a mean score of 96.87 and significantly outperformed Logistic Regression with a mean score of 96.93%. Random Forest lagged behind with an ROC-AUC mean score of 0.909377289. The ROC curves for the top-performing models in average fold exhibit strong concavity toward the upper-left corner, a characteristic indicative of high AUC values.

V. CONCLUSIONS

This research uses several algorithms to analyze educational content, applying classification and recommendation methods to guide students' program selection and help them make more informed decisions based on data-driven outputs suggested by the algorithm. An approach to predict student course placement, focusing on unique features between related undergraduate programs. Across all models, XGBoost emerged as the most effective and reliable classifier for these datasets and tasks. The ensemble approach did not surpass XGBoost and showed greater variability across folds, suggesting limited benefits from aggregation when the constituent models are trained on the same TF-IDF representation, which may yield correlated predictions. A student provides informed information based on the recommendations' results, with directions to make more informed decisions based on the given input data and historical data processed by the ensemble model. This approach gives students who are currently enrolled in the course the opportunity to be informed whether the course is suitable for them or not, which gives them the ability to make an informed decision regarding the course they are currently enrolled in.

VI. AUTHOR'S CONTRIBUTION

Conceptualization: Manocher C. Alipour.

Methodology: Manocher C. Alipour.

Investigation: Manocher C. Alipour.

Discussion of results: Manocher C. Alipour.

Writing – Original Draft: Manocher C. Alipour.

Writing – Review and Editing: Manocher C. Alipour.

Resources: Manocher C. Alipour.

Supervision: Manocher C. Alipour.

Approval of the final text: Manocher C. Alipour.

VII. ACKNOWLEDGMENTS

The authors extend their sincere gratitude to Isabela State University, Angadanan Campus, for their invaluable support and guidance throughout the conduct of this research. Their encouragement and resources were instrumental in the successful completion of this study.

VIII. REFERENCES

- [1] A. N. Hasoon, S. K. Abdulateef, R. S. Abdulameer, and M. L. Shuwandy, "An Intelligent Hybrid AI Course Recommendation Framework Integrating BERT Embeddings and Random Forest Classification," *Computers*, vol. 14, no. 9, p. 353, 2025, doi: 10.3390/computers14090353.
- [2] R. Huang, "Improved content recommendation algorithm integrating semantic information," *J. Big Data*, vol. 10, no. 1, p. 84, 2023, doi: 10.1186/s40537-023-00776-7.
- [3] E. J. Delahoz-Domínguez and R. Hijón-Neira, "Recommender System for University Degree Selection: A Socioeconomic and Standardised Test Data Approach," *Applied Sciences*, vol. 14, no. 18, p. 8311, 2024, doi: 10.3390/app14188311.
- [4] K. Singh, S. Dhawan, and N. Bali, "An Ensemble Learning Hybrid Recommendation System Using Content-Based, Collaborative Filtering, Supervised Learning and Boosting Algorithms," *Autom. Control Comput. Sci.*, vol. 58, no. 5, pp. 491–505, 2024, doi: 10.3103/S0146411624700615.
- [5] H. Gunasekaran, D. Kanmani, and R. Krishnamoorthi, "Optimized Ensemble Learning for Enhanced Crop Recommendations: Leveraging ML for Smarter Agricultural Decision-Making," *Engineering Proceedings*, vol. 82, no. 1, p. 95, 2024, doi: 10.3390/ecs-11-20366.
- [6] L. Paura and I. Arhipova, "Cause Analysis of Students' Dropout Rate in Higher Education Study Program," *Procedia - Soc. Behav. Sci.*, vol. 109, pp. 1282–1286, 2014, doi: <https://doi.org/10.1016/j.sbspro.2013.12.625>.
- [7] R. Y. Chao, "Higher Education in the Philippines BT - International Handbook on Education in Southeast Asia," L. P. Symaco and M. Hayden, Eds. Singapore: Springer Nature Singapore, 2024, pp. 863–889.
- [8] R. Cariaga, "The Philippine Education Today and Its Way Forward (Journal of Ongoing Educational Research)," vol. 1, pp. 40–42, Oct. 2023, doi: 10.5281/zenodo.8418880.
- [9] M. Yuldashev et al., "Collaborative Filtering in Real-Time Library Catalog Recommendations," *Indian J. Inf. Sources Serv.*, vol. 15, pp. 266–274, Dec. 2025, doi: 10.51983/ijiss-2025.IJISS.15.4.30.
- [10] Y. Terui, Y. Inoue, Y. Hamakawa, K. Tatsumura, and K. Kudo, "Collaborative filtering based on nonnegative/binary matrix factorization," *Front. big data*, vol. 8, p. 1599704, Jul. 2025, doi: 10.3389/fdata.2025.1599704.
- [11] M. A. Khan and A. Polyzou, "Session-based Methods for Course Recommendation," Jun. 2024, doi: 10.5281/zenodo.11384740.
- [12] V. A. Nguyen, H.-H. Nguyen, D.-L. Nguyen, and M.-D. Le, "A course recommendation model for students based on learning outcome," *Educ. Inf. Technol.*, vol. 26, no. 5, pp. 5389–5415, 2021, doi: 10.1007/s10639-021-10524-0.
- [13] G. Zhang et al., "Optimizing learning paths: Course recommendations based on graph convolutional networks and learning styles," *Appl. Soft Comput.*, vol. 175, p. 113083, 2025, doi: <https://doi.org/10.1016/j.asoc.2025.113083>.
- [14] C. Troussas, A. Krouska, P. Tselenti, D. Kardaras, and S. Barbounaki, "Enhancing Personalized Educational Content Recommendation through Cosine Similarity-Based Knowledge Graphs and Contextual Signals," *Information*, vol. 14, p. 505, Sep. 2023, doi: 10.3390/info14090505.
- [15] Y. Januzaj and A. Luma, "Cosine Similarity – A Computing Approach to Match Similarity Between Higher Education Programs and Job Market Demands Based on Maximum Number of Common Words," *Int. J. Emerg. Technol. Learn.*, vol. 17, pp. 258–268, Jun. 2022, doi: 10.3991/ijet.v17i12.30375.
- [16] H. Steck, C. Ekanadham, and N. Kallus, "Is Cosine-Similarity of Embeddings Really About Similarity?," in *Companion Proceedings of the ACM Web Conference 2024*, 2024, pp. 887–890, doi: 10.1145/3589335.3651526.
- [17] M. Abdasalam, A. Alzubi, and K. Iyiola, "Student grade prediction for effective learning approaches using the optimized ensemble deep neural network," *Educ. Inf. Technol.*, vol. 30, no. 8, pp. 10159–10183, 2025, doi: 10.1007/s10639-024-13224-7.
- [18] E. Creely and K. Carabott, "Teaching and learning with AI: an Integrated AI-Oriented Pedagogical Model," *Aust. Educ. Res.*, vol. 52, no. 6, pp. 4633–4654, 2025, doi: 10.1007/s13384-025-00913-6.
- [19] N. F. Davar, M. A. Dewan, and X. Zhang, "AI Chatbots in Education: Challenges and Opportunities," *Information*, vol. 16, no. 3, p. 235, 2025, doi: 10.3390/info16030235.
- [20] Y. Fu, Z. Weng, and J. Wang, "Examining AI Use in Educational Contexts: A Scoping Meta-Review and Bibliometric Analysis," *Int. J. Artif. Intell. Educ.*, vol. 35, no. 3, pp. 1388–1444, 2025, doi: 10.1007/s40593-024-00442-w.
- [21] M. S. Kapucu, H. Özcan, A. Aypay, and others, "Predicting Secondary School Students' Academic Performance in Science Course by Machine Learning," *Int. J. Technol. Educ. Sci.*, vol. 8, no. 1, pp. 41–62, 2024.
- [22] S. Wu Fung, S. Tyrväinen, L. Ruthotto, and E. Haber, "ADMM-Softmax: an ADMM approach for multinomial logistic regression," *ETNA - Electron. Trans. Numer. Anal.*, vol. 52, pp. 214–229, Jan. 2020, doi: 10.1553/etna_vol52s214.
- [23] Y. Yao and H. Wang, "Optimal subsampling for softmax regression," *Stat. Pap.*, vol. 60, no. 2, pp. 585–599, 2019, doi: 10.1007/s00362-018-01068-6.
- [24] G. Li, J. Cui, H. Fu, and Y. Sun, "Light GBM and GA Based Algorithm for Predicting and Improving Students' Performance in Higher Education in the Context of Big Data," in *2024 7th International Conference on Education, Network and Information Technology (ICENIT)*, 2024, pp. 1–5, doi: 10.1109/ICENIT61951.2024.00009.
- [25] K. Kesgin, S. Kiraz, S. Kosunalp, and B. Stoycheva, "Beyond Performance: Explaining and Ensuring Fairness in Student Academic Performance Prediction with Machine Learning," *Applied Sciences*, vol. 15, no. 15, p. 8409, 2025, doi: 10.3390/app15158409.
- [26] J. S. Pimentel, R. Ospina, and A. Ara, "Learning time acceleration in support vector regression: a case study in educational data mining," *Stats*, vol. 4, no. 3, pp. 682–700, 2021.

- [27] Y. Chen and L. Zhai, "A comparative study on student performance prediction using machine learning," *Educ. Inf. Technol.*, vol. 28, no. 9, pp. 12039–12057, 2023.
- [28] R. Kadu, P. J. Assudani, T. Mukewar, J. Kapgate, and R. Bijekar, "Student Placement Prediction and Skill Recommendation System using Machine Learning Algorithms," in *2024 International Conference on Inventive Computation Technologies (ICICT)*, 2024, pp. 401–408, doi: 10.1109/ICICT60155.2024.10544738.
- [29] S. Wang, F. Wang, Z. Zhu, J. Wang, T. Tran, and Z. Du, "Artificial intelligence in education: A systematic literature review," *Expert Syst. Appl.*, vol. 252, p. 124167, 2024, doi: <https://doi.org/10.1016/j.eswa.2024.124167>.
- [30] J. Garzón, E. Patiño, and C. Marulanda, "Systematic Review of Artificial Intelligence in Education: Trends, Benefits, and Challenges," *Multimodal Technologies and Interaction*, vol. 9, no. 8, p. 84, 2025, doi: 10.3390/mti9080084.
- [31] D. Dey et al., "The proper application of logistic regression model in complex survey data: a systematic review," *BMC Med. Res. Methodol.*, vol. 25, no. 1, p. 15, 2025, doi: 10.1186/s12874-024-02454-5.
- [32] A. F. Lukman, S. Mohammed, O. Olaluwoye, and R. A. Farghali, "Handling Multicollinearity and Outliers in Logistic Regression Using the Robust Kibria–Lukman Estimator," *Axioms*, vol. 14, no. 1, p. 19, 2025, doi: 10.3390/axioms14010019.
- [33] T. Liu and D. Fan, "Random Forest Algorithm in Information Personalization Recommender System," in *The International Conference on Forthcoming Networks and Sustainability (FoNeS 2022)*, 2022, vol. 2022, pp. 6–10, doi: 10.1049/icp.2022.2367.
- [34] B. M. Hassan and S. M. Elagamy, "Personalized medical recommendation system with machine learning," *Neural Comput. Appl.*, vol. 37, no. 9, pp. 6431–6447, 2025, doi: 10.1007/s00521-024-10916-6.
- [35] Z. Shahbazi and Y. Byun, "Product Recommendation Based on Content-based Filtering Using XGBoost Classifier," *Int. J. Adv. Sci. Technol.*, vol. 29, pp. 6979–6988, Jul. 2020.
- [36] B. Bhavana, J. Karthik, and P. L. Kumari, "A Novel Approach for Product Recommendation using XGBOOST," in *2023 International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT)*, 2023, pp. 256–261, doi: 10.1109/IDCIoT56793.2023.10053453.
- [37] S. Kumar, J. P. Singh, and Shantanu, "Genre-Based Movie Recommender System with XGBoost," in *2024 4th International Conference on Computer, Communication, Control & Information Technology (C3IT)*, 2024, pp. 1–6, doi: 10.1109/C3IT60531.2024.10829451.
- [38] F. Chen and Y. Cui, "Utilizing Student Time Series Behaviour in Learning Management Systems for Early Prediction of Course Performance," *J. Learn. Anal.*, vol. 7, pp. 1–17, Sep. 2020, doi: 10.18608/jla.2020.72.1.
- [39] F. Landi, L. Baraldi, M. Cornia, and R. Cucchiara, "Working memory connections for LSTM," *Neural Networks*, vol. 144, pp. 334–341, 2021.
- [40] Y. Yuan, Y. Zhou, X. Chen, Q. Xiong, and H. C. Okere, "Enhancing Recommendation Diversity and Novelty with Bi-LSTM and Mean Shift Clustering," *Electronics*, vol. 13, no. 19, p. 3841, 2024, doi: 10.3390/electronics13193841.
- [41] J.-W. Tzeng, N.-F. Huang, Y.-H. Chen, T.-W. Huang, and Y.-S. Su, "Personal learning material recommendation system for MOOCs based on the LSTM neural network," *Educ. Technol. Soc.*, vol. 27, no. 2, pp. 25–42, Feb. 2024, [Online]. Available: <https://www.jstor.org/stable/48766161>.
- [42] X. Ni, "User Behavior Analysis and Personalized Recommendation Algorithm in an Online Learning Platform Based on Deep Learning BT - 7th International Conference on Education, Network and Information Technology," 2025, pp. 51–62.
- [43] A. Dey, V. Chauhan, P. Singh, and P. Choudhury, *LSTM-Based Top N Recommendation System using Cognitive Data*. 2022.
- [44] L. Cheng, Y. Yang, K. Zhao, and Z. Gao, "Research and Improvement of TF-IDF Algorithm Based on Information Theory BT - The 8th International Conference on Computer Engineering and Networks (CENet2018)," 2020, pp. 608–616.
- [45] K. K. San, K. E. E. Chaw, and T. T. Nwe, "Attention-Based Neural Networks with TF-IDF Features and BERT Embeddings for Enhanced Course Recommendation System," in *2024 5th International Conference on Advanced Information Technologies (ICAIT)*, 2024, pp. 1–6, doi: 10.1109/ICAIT65209.2024.10754938.
- [46] S. Tiwari, R. Sahu, S. Shripal, and S. Rajpoot, "Innovative Framework for Context-based Recommendation using TF-IDF and Cosine Similarity," *Int. J. Innov. Res. Sci. Technol.*, vol. 12, pp. 9019–9023, Nov. 2023, doi: 10.15680/IJRSET.2023.1206157.
- [47] Z. Pardos and W. Jiang, *Designing for serendipity in a university course recommendation system*. 2020.
- [48] B. K. Saini, "Optimizing Student Academic Performance Prediction Using Heterogeneous Ensemble Learning," *Eur. J. Artif. Intell. Mach. Learn.*, vol. 4, no. 4 SE-Articles, pp. 1–6, 2025, doi: 10.24018/ejai.2025.4.4.77.
- [49] S. A. Alwarthan, N. Aslam, and I. U. Khan, "Predicting student academic performance at higher education using data mining: a systematic review," *Appl. Comput. Intell. Soft Comput.*, vol. 2022, no. 1, p. 8924028, 2022.
- [50] A. Ng and K. Soo, "Random forests," in *Data Science--was ist das eigentlich?! Algorithmen des maschinellen Lernens verständlich erklärt*, Springer, 2018, pp. 117–127.
- [51] A. D. Kumar, R. P. Selvam, and V. Palanisamy, "Hybrid classification algorithms for predicting student performance," in *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, 2021, pp. 1074–1079.
- [52] A. Salini, U. Jeyapriya, and others, "A majority vote based ensemble classifier for predicting students academic performance," *Int. J. Pure Appl. Math.*, vol. 118, no. 24, pp. 1–11, 2018.
- [53] Y. Dong et al., "Deep learning methods may not outperform other machine learning methods on analyzing genomic studies," *Front. Genet.*, vol. Volume 13-2022, 2022, doi: 10.3389/fgene.2022.992070.