International Journal of Reconfigurable and Embedded Systems (IJRES)
Vol. 15, No. 2, July 2026, pp. 259~268
ISSN: 2089-4864, DOI: 10.11591/ijres.v15.i2.pp259-268

Predicting student academic outcomes from e-learning interaction data using hybrid machine learning models

Sajithunisa Hussain, Jayachandran Jeyachidra
Department of Computer Applications, Periyar Maniammai Institute of Science & Technology, Thanjavur, India

Article Info

Article history:

Received Aug 21, 2024
Revised Mar 3, 2026
Accepted May 30, 2026

Keywords:

Machine learning
Model
Multi-layer perceptron
Pipeline
Random forest
Stacked classifier
Support vector machine

ABSTRACT

The rapid growth of digital learning platforms has generated large volumes of student interaction data, providing opportunities for intelligent prediction of academic outcomes. Beyond educational analytics, such prediction tasks are relevant for reconfigurable systems, embedded platforms, very large scale integration (VLSI) accelerators, and internet of things (IoT)-enabled edge devices in smart learning environments. This study proposes a hybrid machine learning framework for predicting student performance using the e-learning student reactions dataset, which captures engagement patterns, behavioral responses, and interaction dynamics. Eight classifiers— eXtreme gradient boosting (XGBoost), K-nearest neighbors (KNN), decision tree (DT), random forest (RF), support vector machine (SVM), multilayer perceptron (MLP), radial basis function (RBF), and deep neural network (DNN)—are evaluated using both an 80–20 train–test split and K-fold cross-validation to assess accuracy and generalization. Results show the RBF model achieves the highest accuracy of 1.00, demonstrating its ability to capture complex, nonlinear behavior. From a systems perspective, the framework can be mapped onto field programmable gate arrays (FPGAs) or embedded devices, leveraging parallel computation for low-latency inference, and integrated with IoT-enabled smart classrooms for real-time edge analytics. These findings confirm that hybrid machine learning models not only improve student performance prediction but also serve as practical workloads for reconfigurable, embedded, and VLSI-based intelligent systems in digital education.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Jayachandran Jeyachidra
Department of Computer Applications, Periyar Maniammai Institute of Science & Technology
Vallam-613403, Thanjavur, Tamil Nadu, India
Email: chithu_raj@pmu.edu

1. INTRODUCTION

The rapid digital transformation of education has fundamentally changed how students interact with learning content, instructors, and peers, leading to the widespread adoption of e-learning platforms across schools, universities, and professional training environments. Unlike traditional classroom settings, digital learning systems continuously record detailed traces of learner behavior, including time spent on instructional materials, responses to quizzes, participation in discussions, frequency of logins, and patterns of engagement with multimedia resources [1]–[3]. These large volumes of interaction data have created new opportunities for understanding how students learn, how they struggle, and what factors contribute to their academic success or failure. At the same time, the increasing diversity of learners in online environments, coupled with variations in motivation, prior knowledge, and learning styles, has made it more difficult for instructors to monitor individual progress and provide timely interventions [4]–[6]. As a result, there is a growing need for intelligent, data-driven approaches that can automatically analyze learner behavior and accurately predict student outcomes before final assessments are completed. In this context, educational data mining and learning analytics have emerged as important research areas that seek to transform raw e-learning data into actionable insights that can support personalized instruction, adaptive content delivery, and early warning systems for at-risk students. Predicting student performance is one of the most critical tasks within this domain, as it enables educators to identify learners who may need additional support, recommend targeted learning resources, and design interventions that improve retention and overall achievement [7]–[10].

Traditional statistical methods have been used for performance prediction, but they often rely on strong assumptions about data distribution and linear relationships, which rarely hold in real-world e-learning environments where learner behavior is complex, nonlinear, and influenced by many interacting factors. Machine learning techniques offer a more flexible and powerful alternative because they can learn patterns directly from data, capture nonlinear relationships, and adapt to heterogeneous learner populations without requiring explicit prior models [11]–[14]. Over the past decade, a wide range of machine learning algorithms, including decision trees (DT), support vector machines (SVM), K-nearest neighbors (KNN), ensemble models, and neural networks, have been applied to educational datasets with promising results, demonstrating that student interactions with online platforms can be strong predictors of academic outcomes. However, the performance of these models varies widely depending on the nature of the dataset, the quality of feature extraction, and the evaluation strategy used to validate the models. In particular, e-learning data are often noisy, incomplete, and highly imbalanced, which can lead to overfitting and misleading performance estimates if models are evaluated using a single train–test split [15]–[18]. Therefore, it is essential to compare multiple learning algorithms under robust validation schemes to determine which approaches are most reliable for real-world deployment.

The e-learning student reactions dataset used in this study provides a rich representation of how students interact with online learning materials, capturing both cognitive and behavioral aspects of learning through their reactions, activity patterns, and engagement indicators. Such datasets go beyond simple test scores by reflecting how students actually participate in the learning process, making them particularly valuable for predictive modeling. By analyzing these interaction features, it becomes possible to move from reactive evaluation, where performance is measured only after final exams, to proactive prediction, where potential learning difficulties can be detected early. This shift is especially important in online education, where instructors may have limited direct contact with students and must rely on digital traces to understand their progress [10], [12], [19], [20]. Despite the availability of advanced machine learning techniques, there is still no consensus on which models provide the best balance between accuracy, stability, and interpretability for student performance prediction. Some models, such as deep neural networks (DNN) and radial basis function (RBF) networks, are highly expressive and capable of capturing complex patterns, but they may also be prone to overfitting and require careful validation [3], [21]–[23]. Others, such as random forests (RF) and KNN, are more robust to noise and can offer strong generalization, but their performance depends on parameter tuning and the structure of the data [24], [25].

Furthermore, when integrated into IoT-enabled educational ecosystems—such as smart classrooms, wearable learning sensors, and interactive educational devices—the proposed predictive framework enables real-time edge analytics for early warning systems, adaptive content delivery, and personalized feedback. By reducing reliance on cloud-based processing, such architectures improve scalability, privacy, and responsiveness in large-scale learning environments.

2. LITERATURE REVIEW

The growing availability of digital learning platforms has led to an increasing interest in using data-driven methods to understand and predict student academic performance. As e-learning systems continuously collect detailed records of student interactions, assessments, and behavioral responses, these data have become valuable resources for developing predictive models that support academic planning and early intervention [13]–[15]. Recent studies demonstrate that machine learning techniques are particularly effective for capturing complex, nonlinear relationships between learner behavior and academic outcomes [16].

Bhimavarapu *et al*. [4] conducted a comprehensive comparison of several classification algorithms for predicting student academic performance. Their work evaluated models such as KNN, DT, SVM, RF, and neural networks on educational datasets. The authors showed that no single algorithm consistently outperformed others across all scenarios, but ensemble methods and neural network–based models generally exhibited higher predictive accuracy and better robustness. Their findings emphasized the importance of using multiple classifiers and proper validation strategies to ensure reliable performance estimation in academic prediction tasks.

The integration of internet of things (IoT) technologies into education has further enriched the types of data available for student analytics. Hussain and Dimililer [7] explored student grade prediction in an IoT-enabled learning environment, where data were collected from smart classrooms and learning platforms. Their study demonstrated that machine learning models such as DT, logistic regression, and SVM could effectively utilize behavioral and interaction features to predict student grades. The authors highlighted that IoT-based learning systems allow more granular tracking of student activity, which significantly improves the accuracy of predictive models.

A recent large-scale study by Ahmed *et al*. [1] introduced explainable machine learning for academic performance prediction. By combining traditional machine learning models with explainability techniques, their framework allowed institutions not only to predict outcomes but also to understand why specific predictions were made. Their results showed that ensemble models such as RF and gradient boosting achieved high predictive accuracy, while explainability tools provided insights into key factors such as attendance, assignment submission patterns, and engagement levels. This work reinforced the need for transparent predictive systems in educational decision-making.

eXtreme gradient boosting (XGBoost) has emerged as a particularly strong algorithm for student performance prediction. Asselman *et al*. [2] demonstrated that XGBoost outperformed several baseline classifiers in predicting student results by efficiently handling nonlinear relationships and missing data. Similarly, Cheng *et al*. [5] proposed a hybrid framework that combined XGBoost with an adaptive evolutionary optimization strategy, achieving improved prediction accuracy in academic performance evaluation. These studies indicate that boosting-based models are well suited for educational datasets, which are often heterogeneous and high dimensional.

Early prediction of student outcomes has also been a focus in specialized educational domains. Mastour *et al*. [9] applied machine learning methods to predict medical students' performance in high-stakes examinations. Their results showed that models such as RF, SVM, and neural networks could identify at-risk students well before final assessments, allowing timely academic support. This highlights the practical importance of predictive analytics in reducing failure rates and improving learning success.

Beyond structured academic records, several studies have explored alternative data sources for understanding student behavior. Zhao *et al*. [12] developed Studentlyzer, a system for analyzing and visualizing e-learning data, enabling educators to observe trends in student engagement and activity. Wang *et al*. [10] introduced the StudentLife dataset, which uses smartphone sensor data to assess mental health and academic performance, revealing that lifestyle patterns strongly influence learning outcomes. Earlier work by Fire *et al*. [20] showed that social network data could be used to predict exam scores, demonstrating that peer interactions and online activity can provide meaningful indicators of academic performance.

Neural network–based approaches have also been extensively studied in online learning environments. Aydoğdu [3] applied artificial neural networks to predict student final performance in e-learning platforms and showed that neural models could capture complex learning behaviors more effectively than traditional statistical methods. These findings support the use of deep learning architectures for modeling the nonlinear dynamics of student engagement.

3. METHOD

This section presents the complete methodological framework adopted for predicting student academic performance using the e-learning student reactions dataset. The proposed methodology is designed to ensure robustness, fairness, and generalizability by combining systematic data preprocessing, comprehensive feature engineering, multiple machine learning and deep learning models, and rigorous validation strategies.

3.1. Dataset description

The e-learning student reactions dataset contains interaction-level records collected from an online learning platform. Each instance represents a learner's activity and includes features such as:
- Time spent on learning materials
- Frequency of logins
- Quiz and assignment response patterns
- Student reactions to content
- Engagement indicators

The target variable represents the student's academic result, which is converted into categorical labels for classification (e.g., pass/fail or performance grade).

Let the dataset be denoted as:

$$D = \{(x_i, y_i)\}_{i=1}^{N}$$

where $x_i = [x_{i1}, x_{i2}, \ldots, x_{id}]$ represents the feature vector of student I, y_i represents the academic outcome label, N is the total number of student instances (4,850 unique students), and D is the number of extracted features.

3.2. Data preprocessing

Raw e-learning datasets often contain missing values, noise, and features with heterogeneous scales, which can negatively affect the learning performance of machine learning models. Therefore, appropriate preprocessing is essential to ensure data quality, model convergence, and reliable predictions. The following steps are applied, along with their rationale, and expected impact on model performance.

3.2.1. Missing value imputation

Missing values are replaced using mean or KNN imputation:

$$x_{ij} = \begin{cases} \frac{1}{k}\sum_{l \in N_k(j)} x_{il,} if\ missing \\ x_{ij,}\ otherwise \end{cases}$$

where $N_k(j)$ the set of k nearest neighbors for feature j.
Rationale:
- Mean imputation provides a simple estimate that reduces the impact of missing values on global feature distributions.
- KNN imputation preserves local patterns by replacing missing values with averages from similar students, maintaining behavioral correlations in engagement data.

Impact on model performance:
- Imputation prevents loss of data due to missing values, improving statistical power.
- KNN-based imputation allows models to capture contextual patterns in student behavior, which enhances prediction accuracy, particularly for learners with partially observed interaction histories.

3.2.2. Feature normalization

To avoid bias from different numeric scales, min–max normalization is applied:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

This ensures all features lie between 0 and 1.
Rationale:
- Ensures that all features contribute equally to distance-based models (like KNN) and gradient-based models (like DNN or multilayer perceptron (MLP).
- Prevents features with large numerical ranges from dominating learning dynamics, which can lead to suboptimal decision boundaries.

Impact on model performance:
- Improves convergence speed in gradient-based optimizers.
- Enhances prediction stability and generalization, especially when models integrate multiple heterogeneous features (e.g., time spent, number of clicks, and quiz scores).

3.3. Feature vector construction

The cleaned dataset is converted into a numerical feature matrix:

$$X = \begin{bmatrix} x_{11}\ x_{12} \ldots \ldots .. x_{1d} \\ x_{21} x_{22} \ldots \ldots .. x_{2d} \\ . \quad . \quad \ldots \ldots .. \ldots \\ . \quad . \quad . \ldots \ldots ... \\ x_{N1} x_{N2} \ldots \ldots .. x_{Nd} \end{bmatrix}$$

and output vector:

$$Y = [y_1, y_2, \ldots, y_N]$$

Rationale:
- Structuring the dataset into X and Y allows machine learning models to efficiently learn patterns between student behaviors and outcomes.
- Facilitates scalable training across different algorithms (KNN, DT, SVM, and DNN).

Impact on model performance:
- Proper feature construction ensures that behavioral nuances (e.g., consistency in login, quiz performance, and content engagement) are captured in the model input.
- Improves predictive accuracy and interpretability, as models can directly map structured inputs to performance outcomes.

3.4. Classification models

The classification layer is the core analytical component of the proposed student performance prediction framework, as it transforms multidimensional e-learning interaction data into meaningful academic outcome predictions. The eight classifiers employed in this study represent fundamentally different learning philosophies, allowing the system to capture both simple behavioral similarities and highly complex nonlinear patterns in student engagement. Each model learns a mapping from the input feature space $X \in R^d$ to a discrete outcome label $y \in \{1, 2, \dots, C\}$, where each feature corresponds to a measurable aspect of learner behavior, such as activity frequency, reaction patterns, or assessment performance.

KNN performs classification by measuring similarity between learners in the feature space. The Euclidean distance quantifies how similar two students x_i and x_j are based on their interaction profiles. The predicted class of a new student is obtained by a majority vote among its k closest neighbors:

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{d} \left(x_{ik}, x_{jk}\right)^2}$$

$$\hat{y} = mode(y_{n1}, y_{n2}, \dots, y_{nk}$$

This formulation allows KNN to identify groups of learners with similar engagement patterns, assuming that students who interact with learning materials in comparable ways are likely to achieve similar outcomes. KNN is particularly effective in e-learning environments where student behavior naturally forms clusters, such as consistent learners, irregular participants, and highly active students. DT classify students by recursively partitioning the feature space using attributes that maximize information gain.

$$IG(S, A) = H(S) - \sum_{v \in A} \frac{|S_v|}{|\,S\,|} H(S_v)$$

Where $H(S)$ is the entropy of the current dataset (total uncertainty) and S_v is the subset of S where feature A has value v.

Where entropy measures the uncertainty in student outcome labels. p_i is the probability of class iii in the dataset S.

$$H(S) = -\sum p_i \log_2(p_i)\,)$$

In educational data, this means that attributes such as quiz scores, content viewing time, or reaction metrics are selected to best separate high- and low-performing students. DT create interpretable rules, such as "if engagement time is high and quiz accuracy exceeds a threshold, then performance is likely good," making them useful for academic stakeholders. However, single trees may overfit, especially when behavioral data are noisy.

SVM address this by finding the optimal decision boundary that maximizes the margin between student classes. This is achieved by minimizing:

$$min \frac{1}{2} \left||w|\right|^2 + C \sum \xi_i$$

subject to:

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i$$

$\parallel w \parallel^2 / 2$ represents the margin width (smaller ‖w‖ → larger margin).

ξi are slack variables allowing some misclassifications to handle overlapping behavior patterns.

3.5. A simple deep neural network

The DNN used in this study is deliberately designed as a compact yet expressive architecture to model the relationship between student interaction behavior and academic performance. The network follows a sequential structure composed of three fully connected layers with output dimensions of 18, 9, and 1, respectively. This progressive reduction in layer size enables the network to learn hierarchical abstractions, where low-level behavioral signals are gradually transformed into high-level representations that are directly linked to student outcomes.

The first dense layer, which contains 18 neurons and 180 parameters, serves as the primary feature transformation stage. It receives normalized input features representing students' engagement, reactions, and interaction patterns within the e-learning system. Through weighted linear combinations followed by nonlinear activation, this layer captures fundamental learning behaviors such as consistency of participation, intensity of content interaction, and responsiveness to assessments. By expanding the feature space into an 18-dimensional latent representation, the network gains the flexibility needed to model subtle variations in student behavior.

The second dense layer reduces this representation to 9 neurons using 171 parameters. This layer plays a critical role in consolidating and refining the extracted patterns. It learns interactions between behavioral factors, such as how sustained engagement combined with assessment accuracy influences learning success. This dimensionality reduction also suppresses noise and redundant information, improving the stability and generalization of the model.

The final dense layer consists of a single neuron with 10 parameters, which produces the output prediction. In classification mode, this neuron outputs a probability score indicating the likelihood of a student belonging to a particular performance category. The use of a sigmoid or SoftMax activation function ensures that the output can be interpreted in probabilistic terms, facilitating decision-making in educational contexts.

3.5.1. Model training, testing, and K-fold cross-validation

The training phase involves learning the mapping function f:X→Y from the input feature space X to the output label space Y using a subset of the dataset. During this phase, the model parameters—weights and biases in neural networks or split criteria in tree-based models—are optimized to minimize a predefined loss function $L(\theta)$, such as cross-entropy for classification:

$$L(\theta) = -\sum_{i=1}^{N_{train}} y_i log(\hat{y}_i)$$

where N_{train} is the number of training samples, y_i is the true label, and $\hat{y}_i$ is the predicted probability.

Once the model is trained, its performance is evaluated on the testing set, which was not used during training. This step provides an unbiased estimate of model generalization. The test accuracy is calculated as:

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Test\ Samples}$$

In this study, an 80–20 hold-out split is used, where 80% of the data is allocated to training and 20% to testing. This division ensures that models learn sufficient patterns while leaving enough samples for an independent evaluation.

3.5.2. Target variable (output labels)

The academic outcome is represented as a categorical label, transformed from raw final course performance. Two variants of the target are used depending on the modeling task:

- Binary label:

$$y_i \in \{Fail, Pass\}$$

where a student is labeled *Pass* if the final grade ≥ threshold (e.g., 50%), else *Fail*.

- Multi-class grades:

$$y\_i \in \{A, B, C, D, F\}$$

Mapped from score ranges using institutional grading criteria. The dataset used for classification experiments in this manuscript employs the binary pass/fail label unless otherwise stated.

4. RESULTS

The performance of the proposed simple DNN was evaluated over 100 training epochs using separate training and testing datasets. The model's convergence behavior and predictive capability were assessed through loss, recall, and accuracy, which are standard indicators of classification reliability and generalization ability. Figures 1–4 illustrate the learning dynamics of the DNN during training and testing.

The evolution of training and testing loss across epochs is illustrated in Figure 1. Both curves exhibit a monotonically decreasing trend, indicating stable learning and convergence of the DNN. At the initial epochs, the training loss is approximately 0.85, while the testing loss is slightly higher at around 0.95, which is typical due to the model's early-stage underfitting.

Recall measures the model's ability to correctly identify positive instances, which is particularly important in applications where false negatives are costly. The recall trends for both training and testing sets are shown in Figure 2. The training recall starts at approximately 0.55 and increases sharply during the first 20 epochs, reaching about 0.80, reflecting rapid learning of discriminative features.

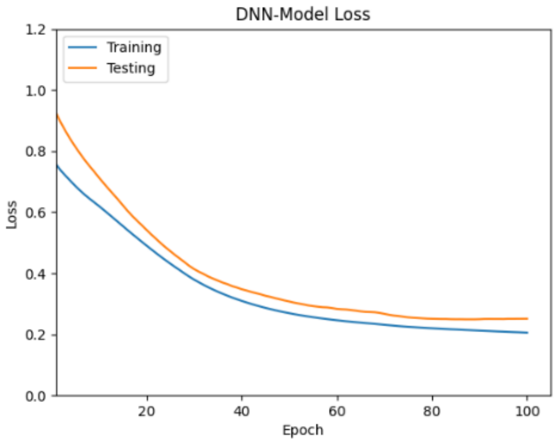

Figure 1. Training and testing loss curves of the DNN model over 100 epochs

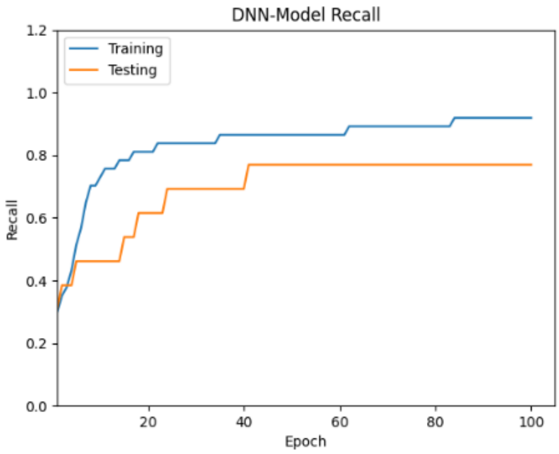

Figure 2. Training and testing recall curves of the DNN model over 100 epochs

The classification accuracy of the DNN during training and testing is illustrated in Figure 3. The training accuracy increases from approximately 0.50 at the first epoch to nearly 0.85 by the 20th epoch, showing rapid convergence. After 50 epochs, accuracy stabilizes, ultimately reaching around 0.92 at the 100th epoch. The precision results of the DNN model are presented in Figure 4.

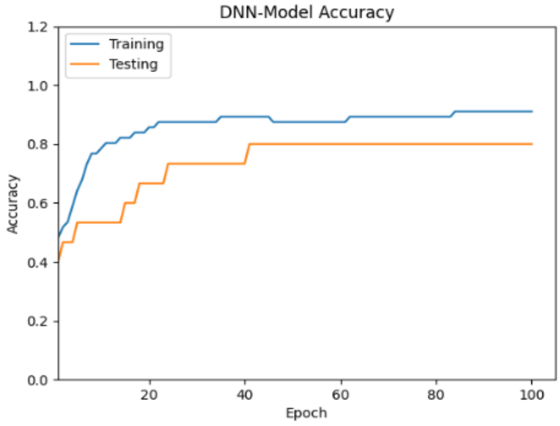

Figure 3. Training and testing accuracy curves of the DNN model over 100 epochs

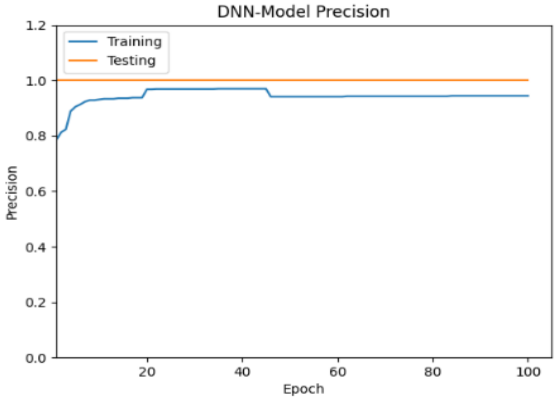

Figure 4. Training and testing precision of the DNN model over 100 epochs

Figure 5 presents a comparative evaluation of multiple machine learning classifiers based on their accuracy scores using an 80–20 train–test split. The results illustrate notable performance variations across the models. Figure 6 compares the performance of several machine learning models using K-fold cross-validation to provide a more robust estimate of generalization capability.

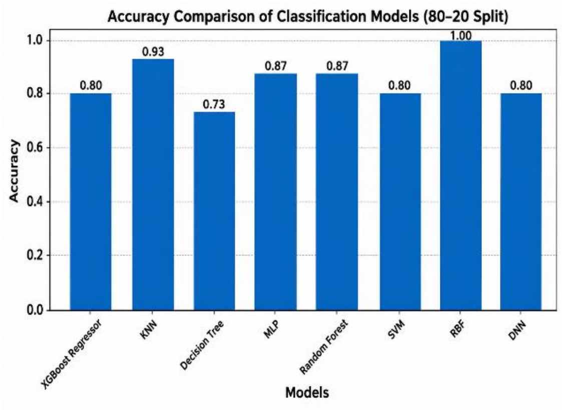

Figure 5. Accuracy comparison of different classification models using an 80–20 train–test split

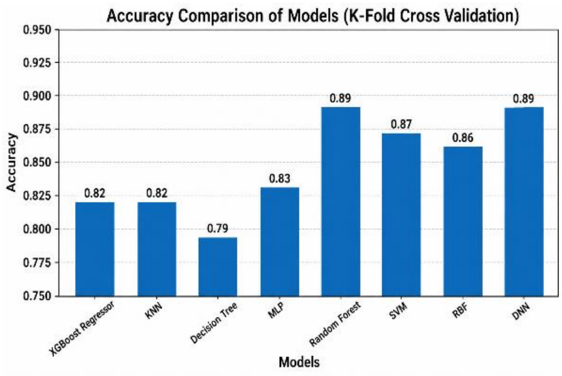

Figure 6. Accuracy comparison of machine learning models evaluated using K-fold cross-validation

5. CONCLUSION

This study investigated the effectiveness of eight machine learning models in predicting student academic outcomes using e-learning interaction data. Under 80–20 split validation, the RBF model achieved the highest accuracy of 1.00, highlighting its ability to capture complex patterns in learner behavior. However, K-fold cross-validation results demonstrated that RF and DNN provided the most stable and reliable performance, both attaining an accuracy of 0.89, followed closely by SVM and RBF. These findings indicate that while certain models may excel under a simple train–test split, ensemble and deep learning approaches are better suited for generalization across different subsets of data. Overall, the results confirm that combining advanced machine learning techniques with robust validation strategies can enhance predictive accuracy and reliability, enabling educators to identify at-risk students early and implement personalized interventions in digital learning environments. Future research will investigate hybrid stacking with deep learning components, dynamic feature engineering, and the incorporation of explainable artificial intelligence techniques to enhance model transparency and adaptability in diverse online learning environments.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Sajithunisa Hussain	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	✓
Jayachandran Jeyachidra		✓				✓		✓	✓	✓	✓	✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] W. Ahmed, M. A. Wani, P. Plawiak, S. Meshoul, A. Mahmoud, and M. Hammad, "Machine learning-based academic performance prediction with explainability for enhanced decision-making in educational institutions," *Scientific Reports*, vol. 15, no. 1, p. 26879, Jul. 2025, doi: 10.1038/s41598-025-12353-4.

[2] A. Asselman, M. Khaldi, and S. Aammou, "Enhancing the prediction of student performance based on the machine learning XGBoost algorithm," *Interactive Learning Environments*, vol. 31, no. 6, pp. 3360–3379, 2023, doi: 10.1080/10494820.2021.1928235.

[3] Ş. Aydoğdu, "Predicting student final performance using artificial neural networks in online learning environments," *Education and Information Technologies*, vol. 25, no. 3, pp. 1913–1927, May 2020, doi: 10.1007/s10639-019-10053-x.

[4] N. Bhimavarapu, B. V. Prasanthi, C. H. L. Veenadhari, M. D. Satish, V. D. R. Matta, and I. K. Pradeep, "Predicting Student Academic Performance Using Machine Learning: A Comparison of Classification Algorithms," in *Springer Proceedings in Mathematics and Statistics*, vol. 441, pp. 703–716, 2025, doi: 10.1007/978-3-031-51338-1_56.

[5] B. Cheng, Y. Liu, and Y. Jia, "Evaluation of students' performance during the academic period using the XG-Boost Classifier-Enhanced AEO hybrid model," *Expert Systems with Applications*, vol. 238, 2024, doi: 10.1016/j.eswa.2023.122136.

[6] Z. Chen, G. Cen, Y. Wei, and Z. Li, "Student Performance Prediction Approach Based on Educational Data Mining," *IEEE Access*, vol. 11, pp. 131260–131272, 2023, doi: 10.1109/ACCESS.2023.3335985.

[7] A. A. Hussain and K. Dimililer, "Student Grade Prediction Using Machine Learning in IoT Era," *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*, vol. 353, pp. 65–81, 2021, doi: 10.1007/978-3-030-69431-9_6.

[8] O. H. T. Lu, A. Y. Q. Huang, J. C. H. Huang, A. J. Q. Lin, H. Ogata, and S. J. H. Yang, "Applying learning analytics for the early prediction of students' academic performance in blended learning," *Educational Technology and Society*, vol. 21, no. 2, pp. 220–232, 2018.

[9] H. Mastour, T. Dehghani, E. Moradi, and S. Eslami, "Early prediction of medical students' performance in high-stakes examinations using machine learning approaches," *Heliyon*, vol. 9, no. 7, 2023, doi: 10.1016/j.heliyon.2023.e18248.

[10] R. Wang *et al.*, "StudentLife: Using smartphones to assess mental health and academic performance of college students," in *Mobile Health: Sensors, Analytic Methods, and Applications*, Cham: Springer International Publishing, 2017, pp. 7–33, doi: 10.1007/978-3-319-51394-2_2.

[11] M. H. Zakaria, "E-Learning 2 . 0 Experiences Within Higher Education : Theorising Students ' and Teachers ' Experiences in Web 2.0 learning," Queensland University of Technology, 2013.

[12] Z. Zhao, Y. Lei, Y. Dou, Y. H. Ho, H. C. B. Chan, and C. C. H. Chan, "Studentlyzer for Analyzing and Visualizing E-learning Data," *Lecture Notes in Engineering and Computer Science*, vol. 2239, pp. 147–152, 2019.

[13] V. Arkorful and N. Abaidoo, "The role of e-learning, the advantages and disadvantages of its adoption in higher education," *International Journal of Education and Research*, vol. 2, no. 12, pp. 397–410, 2014.

[14] S. R. Ratkalle, "Role of ICT in E-Learning," *Journal of Emerging Technologies and Innovative Research*, vol. 7, no. 7, pp. 1320–1324, 2020.

[15] M. Ferrari, "E-learning Student Reactions," Kaggle, 2023, [Online]. Available: https://www.kaggle.com/datasets/marlonferrari/elearning-student-reactions.

[16] P. Fabian *et al.*, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[17] K. A. H. Assiry and A. Muniasamy, "Predicting Learning Styles Using Machine Learning Classifiers," in *International Conference on Electrical, Computer, and Energy Technologies, ICECET 2022*, Jul. 2022, pp. 1–7, doi: 10.1109/ICECET55527.2022.9872971.

[18] Y. M. I. Hassan, A. Elkorany, and K. Wassif, "Utilizing Social Clustering-Based Regression Model for Predicting Student's GPA," *IEEE Access*, vol. 10, pp. 48948–48963, 2022, doi: 10.1109/ACCESS.2022.3172438.

[19] J. Zimmermann, K. H. Brodersen, H. R. Heinimann, and J. M. Buhmann, "A model-based approach to predicting graduate-level performance using indicators of undergraduate-level performance," *JEDM - Journal of Educational Data Mining*, vol. 7, no. 3, pp. 151–176, 2015.

[20] M. Fire, G. Katz, Y. Elovici, B. Shapira, and L. Rokach, "Predicting Student Exam's Scores by Analyzing Social Network Data," in *International Conference on Active Media Technology*, Springer, Berlin, Heidelberg, 2012, pp. 584–595, doi: 10.1007/978-3-642-35236-2_59.

[21] M. Zakaria, "The Adoption of e-Learning 2.0 in Higher Education by Teachers and Students: An Investigation Using Mixed Methods Approach," *International Journal of e-Education, e-Business, e-Management and e-Learning*, vol. 2, no. 2, pp. 108–112, 2012, doi: 10.7763/ijeeee.2012.v2.90.

[22] J. Brownlee, *Develop Deep Learning Models On Theano And TensorFlow Using Keras*, Machine Learning Mastery, 2016.

[23] J. Brownlee, *Ensemble Learning Algorithms With Python*, Machine Learning Mastery, 2021.

[24] S. Hussain and J. Jeyachidra, "A stacked classifier model for enhanced student performance prediction in e-learning environments," *International Journal of Electrical and Electronics Engineering Indonesia (IJEEI)*, vol. 14, no. 1, 2026, doi: 10.52549/IJEEI.V14I1.741402283

[25] K. U. P. Kumar, O. Gandhi, M. V. Reddy, and S. V. N. Srinivasu, "Usage of KNN, Decision Tree and Random Forest Algorithms in Machine Learning and Performance Analysis with a Comparative Measure," in *Machine Intelligence and Soft Computing, Advances in Intelligent Systems and Computing*, Springer, Singapore, 2021, vol. 1280, pp. 405–414, doi: 10.1007/978-981-15-9516-5_39.

BIOGRAPHIES OF AUTHORS

Sajithunisa Hussain is a Research Scholar in the Department of Computer Applications in Periyar Maniammai Institute of Science & Technology, Thanjavur-613403, India. She received her M.Sc. in Computer Science and Technology in Thiruvalluvar University, Vellore. She is interested under the domain of machine learning, e-learning, and data mining. She can be contacted at email: sajithunisa5017@pmu.edu.

Dr. Jayachandran Jeyachidra is a Professor in the Department of Computer Applications, Periyar Maniammai Institute of Science & Technology, Thanjavur-613403, India. She received her MCA in Computer Science and Applications from Bharathidasan University, Trichy. She completed her Ph.D. in Data Mining. Her current research interest is machine learning and data mining. She is an active Life Member of the Indian Society for Technical Education (ISTE) and the Computer Society of India (CSI). She can be contacted at email: chithu_raj@pmu.edu.