International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 5, Sep-Oct-2026, ISSN (Online): 2395-566X

Student Performance Prediction Using Machine Learning

Dinesh G[1], Kabilan KM[2], Sowndar K[3], Vikram S[4] Indhuja N[5]
Department of Artificial Intelligence and Data Science, Sengunthar Engineering College, Tiruchengode, Tamil Nadu, India

Abstract - Student performance prediction is an important application of machine learning in educational data mining. Academic institutions collect different types of information, including attendance, internal assessment marks, previous academic performance, study habits, and assignment completion. Analyzing these factors manually can be time-consuming, particularly when the number of students increases. This paper presents a machine-learning-based framework for predicting student academic performance using relevant educational attributes. The proposed system consists of data collection, data preprocessing, exploratory data analysis, feature selection, model training, evaluation, and prediction. Classification algorithms such as Logistic Regression, Decision Tree, Random Forest, and Support Vector Machine can be evaluated to identify a suitable model for categorizing students into performance levels. Regression methods can alternatively be used when the target variable is a numerical examination score. The system is intended to help educators identify students who may require additional academic support and plan appropriate interventions. Model performance is evaluated using appropriate classification or regression metrics. Since actual experimental measurements depend on the selected dataset and implementation, the proposed framework does not assume unverified accuracy values. The study also discusses data privacy, fairness, interpretability, and the limitations of predictive educational systems.

Keywords-Artificial Intelligence (AI), Automated Policing, Law Enforcement Technology, Public Safety, Predictive Policing, Ethical Automation.

I. INTRODUCTION

Education plays an essential role in individual development and societal progress. Academic institutions continuously attempt to improve learning outcomes by monitoring student progress and identifying difficulties at an early stage. However, conventional academic evaluation often depends on examination marks, manual record analysis, and periodic assessments. These approaches may not always reveal the combination of factors that influence a student's academic performance.

The increasing availability of educational data provides opportunities to apply machine learning techniques to academic analysis. Educational data may include attendance percentage, previous examination scores, internal assessment marks, assignment completion, study time, and other relevant learning indicators. Machine learning algorithms can analyze relationships among these variables and learn patterns associated with academic outcomes.

Student performance prediction aims to estimate a student's likely academic outcome using historical and current educational information. Depending on the application, the target can be a numerical score, a pass or fail category, or a

performance group such as low, medium, and high. Predictive results can support teachers in identifying students who may benefit from additional instruction, mentoring, or learning resources.

Educational data mining investigates methods for discovering useful patterns in educational environments. Romero and Ventura reviewed important data-mining approaches and applications in education [1]. Subsequent work has examined predictive modeling, student behavior, and learning analytics as ways to understand academic processes [2], [3].

This paper proposes a structured machine-learning framework for student performance prediction. The framework is designed to compare multiple algorithms using a consistent preprocessing and evaluation procedure. Rather than treating a prediction as a final judgment about a student, the system is intended to provide decision-support information for educators.

© 2026 IJSRET

The major contributions of this work are as follows:

- A structured pipeline for preparing educational data for predictive analysis.
- A framework for comparing classification algorithms under consistent evaluation conditions.
- A set of evaluation metrics for measuring predictive performance beyond accuracy alone.
- A discussion of practical implementation, interpretability, fairness, and responsible use in educational institutions.

The remainder of this paper describes related work, the problem definition, the proposed methodology, the evaluation framework, the discussion of potential applications, and the conclusion.

II. RELATED WORK

Educational data mining has developed as a research area focused on extracting knowledge from educational environments. Romero and Ventura [1] discussed the use of datamining methods in educational systems, including classification, clustering, and prediction. Their review highlights the potential of data analysis for understanding learner behavior and supporting educational decisions.

Baker and Inventado [2] described educational data mining as an approach for analyzing information generated during learning activities. Such information may be used to investigate patterns that are difficult to identify through manual inspection alone. The selection of features andalgorithms depends on the educational problem, the available data, and the intended use of the results.

Siemens and Long [3] discussed the role of analytics in education and the opportunities created by collecting and analyzing learning data. Predictive analytics can assist institutions in monitoring progress, although the quality of the underlying data remains essential.

Different machine-learning algorithms are suitable for student performance prediction. Logistic Regression provides a relatively interpretable baseline for classification. Decision Trees represent decisions through a sequence of featurebased rules. Random Forest combines predictions from multiple decision trees to reduce dependence on a single tree [4]. Support Vector Machines can construct decision boundaries in high-dimensional feature spaces [5].

The choice of an algorithm should not be based only on its theoretical properties. Dataset size, class imbalance, missing values, feature types, and the intended prediction target can affect the final outcome. A model that performs well on one dataset may not generalize to another institution or student population.

This work organizes these techniques into a common experimental framework. Its focus is on reproducible preprocessing, appropriate evaluation, and responsible interpretation rather than claiming that one algorithm is universally superior.

III. PROBLEM STATEMENT

Educational institutions may have difficulty identifying students who need academic support before final examinations. Attendance records, assessment marks, assignment submissions, and study habits may be stored separately, making it difficult to analyze them collectively.
Manual analysis can require substantial effort and may not consistently identify patterns across a large student population. Furthermore, relying on a single indicator, such as attendance or a previous examination score, may overlook other relevant factors.

The problem addressed in this study is to develop a machine-learning framework that uses available educational attributes to estimate student performance. The framework should support the following objectives:

- Collect and organize relevant student academic data.
- Handle missing values and inconsistent data.
- Identify features suitable for prediction.
- Train and compare appropriate machine-learning models.
- Evaluate the models using suitable performance metrics.
- Present predictions in a form that supports academic intervention and further review.

© 2026 IJSRET

The prediction must be understood as an estimate rather than a guaranteed outcome. Student achievement is influenced by multiple factors, including learning opportunities, personal circumstances, teaching methods, and changes over time. A predictive system cannot fully represent every influence.

IV. OBJECTIVES AND SCOPE

The primary objective is to design a predictive framework that can analyze student-related features and estimate academic performance.

The specific objectives are:

- To preprocess educational records into a consistent format suitable for machine learning.
- To explore relationships between input features and academic outcomes.
- To compare classification models using a common training and testing procedure.
- To determine which features are associated with predictions, where the selected model supports interpretation. • To support early identification of students who may benefit from additional academic assistance.

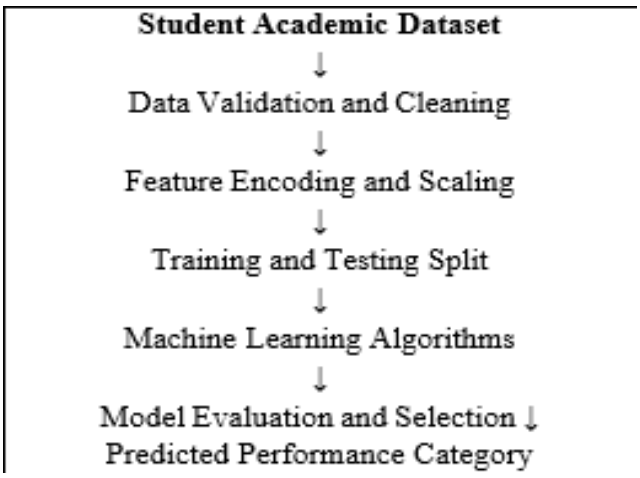
The scope of the proposed system includes structured educational data and supervised machine-learning methods. It does not replace teachers, examinations, counseling, or institutional assessment procedures. Its usefulness depends on the reliability, representativeness, and appropriate handling of the data.

V. PROPOSED SYSTEM

The proposed system consists of six major stages: data collection, preprocessing, feature preparation, model training, model evaluation, and prediction. Each stage is designed to reduce data-related errors and improve the reliability of the final output.

1. System Architecture

The overall workflow is represented as follows:



be performed using training data and cross-validation where appropriate.

After evaluation, the selected model can receive new student records and produce an estimated score or performance category. The output should be presented with an explanation of the prediction's limitations and, where possible, the factors contributing to the model output.

Functional Modules

- Data Collection: Collect historical academic records from an authorized educational dataset. Each record represents a student or an appropriate student-assessment observation.
- Data Preprocessing: Identify missing values, duplicate records, inconsistent labels, and unsuitable attributes. Apply transformations using training data to avoid information leakage.
- Feature Preparation: Select input variables relevant to the target outcome. Encode categorical features and scale numerical features when required by the chosen algorithm.
- Model Training: Train candidate algorithms on the training data. Use a validation strategy to select suitable hyperparameters and compare model behavior.
- Evaluation: Evaluate predictions on data not used to fit the model. Report multiple metrics to reveal different types of errors.
- Prediction: Use the selected model to estimate academic performance for new records. Predictions should be

© 2026 IJSRET

reviewed alongside academic context rather than interpreted in isolation.

VI. DATASET DESCRIPTION

The proposed framework can use a publicly available educational dataset or an institutionally approved dataset

1. Data Preprocessing Let the containing appropriate student records. The exact dataset, sample size, collection period, and feature definitions must be documented when the implementation is finalized.
2. Potential input attributes are presented in Table I. Their inclusion depends on actual dataset availability and the intended prediction task.
3. The target variable must be defined before model training. For classification, a numerical score can be converted into categories using predefined academic criteria. For regression, the original numerical score can be retained as the target.

For example, performance categories may be defined as low, medium, and high. However, the threshold values must follow the applicable institutional grading system or a documented research definition. Arbitrary thresholds can change the class distribution and affect evaluation results.

Table I

Potential Features For Student Performance Prediction

Feature	Description
Attendance	Percentage of classes
Internal Marks	Scores obtained in internal assessments
Previous Score	Earlier examination performance
Study Time	Reported or recorded study duration
Assignments	Assignment completion or scores
Participation	Available learning activity indicators
Target Variable	Final score or performance category

Data Quality

- The dataset should be checked for missing values, inconsistent formats, duplicate observations, and impossible values. For example, an attendance percentage outside the valid range of 0 to 100 requires investigation.
- Sensitive personal information should be excluded unless it is necessary, authorized, and appropriately protected. Student identifiers should be removed or replaced with nonidentifying keys when they are not required for analysis.
- The dataset should also be checked for class imbalance. If only a small proportion of observations belong to a particular performance category, accuracy alone may not adequately describe the model's behavior.

VII. METHODOLOGY

The methodology consists of data preprocessing, feature hyperparameter selection, and final evaluation

Data Preprocessing Let the

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

. where N is the number of observations, xi represents the input feature vector, and yi is the corresponding target. The input vector can be written as

$$x_i = [x_{i1}, x_{i2}, \ldots, x_{ip}], \quad (2)$$

where p is the number of input features.

Missing numerical values may be imputed using a statistic such as the training-set median. Categorical missing values can be handled using a suitable category or a training-set mode. The selected approach should be documented and applied consistently.

Categorical features can be converted into numerical representations using one-hot encoding or another appropriate encoding strategy. Numerical scaling may be necessary for algorithms that depend on feature magnitudes.

For a numerical feature x, standardization can be expressed

© 2026 IJSRET

As

For a numerical feature x, standardization can be expressed as

$$z = \frac{x - \mu_{train}}{\sigma_{train}}, \quad (3)$$

where μ_{train} and σ_{train} are the mean and standard deviation calculated from the training data. The same values must be used to transform validation, test, and new observations.
Preprocessing parameters must not be estimated from the complete dataset before splitting, because doing so can allow information from the test set to influence model development.

Feature Selection

Feature selection aims to identify variables that provide useful information for the prediction task. Relevant features may include attendance, previous scores, assessment marks, and assignment completion, depending on the dataset.
Highly redundant features may increase model complexity without contributing substantial additional information. Correlation analysis and model-based feature importance can help investigate relationships among features.
Feature selection must be performed without using heldout test outcomes. A feature should also be checked for target leakage. For example, including a final examination score as an input when predicting that same final examination score would make the task invalid.

Dataset Splitting

The dataset should be divided into training, validation, and testing subsets when the available sample size permits. A possible starting configuration is 70% training data, 15% validation data, and 15% testing data.

For classification, stratified splitting can help preserve class proportions across subsets. If observations are repeated for the same student, records from one student should not be distributed across training and testing data when doing so would create an unrealistic evaluation.

When the task predicts future performance, a chronological split may be more appropriate than a random split. The partitioning strategy should reflect the intended deployment scenario.

Logistic Regression

Logistic Regression is a supervised classification algorithm that estimates class probabilities. For binary classification, the probability of the positive class is given by

$$P(y = 1 \mid x) = \frac{1}{1 + \exp[-(\beta_0 + \beta^T x)]}, \quad (4)$$

where β_0 is the intercept and β represents the model coefficients.
The estimated probability can be converted into a class prediction using a decision threshold. Logistic Regression provides a useful baseline and can support interpretation of the relationship between input variables and predictions. Multiclass problems can be handled through suitable multiclass extensions.

Decision Tree

A Decision Tree recursively divides observations according to feature-based rules. Each internal node represents a decision, while leaf nodes represent predicted classes or numerical values.

For classification, the Gini impurity of a node is

$$G = 1 - \sum_{k=1}^{K} p_k^2, \quad (5)$$

where p_k is the proportion of observations belonging to class k and K is the number of classes.
The tree selects splits that improve node purity according to its splitting criterion. Decision Trees are relatively easy to interpret, but an unrestricted tree can overfit the training data. Maximum depth, minimum leaf size, and pruning can be used to control complexity.

Random Forest

© 2026 IJSRET

Random Forest is an ensemble learning algorithm that combines predictions from multiple decision trees. Each tree is trained using a bootstrap sample, and candidate features are considered at the splitting stages.

For classification, the predicted class can be obtained through majority voting:

$$\hat{y} = \text{mode}\{T1(x), T2(x), ..., TM(x)\}, \quad (6)$$

where $Tj(x)$ is the prediction from tree j and M is the number of trees.

For probability-based classification, the estimated class probabilities can instead be averaged across trees. Random Forest can model nonlinear relationships and interactions between features. However, its performance depends on the data and model configuration, and its predictions may be less directly interpretable than those of a small decision tree.

Support Vector Machine

A Support Vector Machine (SVM) seeks a decision boundary that separates classes while maximizing the margin between them. A linear decision function can be represented as

$$f(x) = w^T x + b, \quad (7)$$

where w is the weight vector and b is the bias.

Kernel functions allow SVMs to represent nonlinear decision boundaries. Numerical feature scaling is generally important for this algorithm. Hyperparameters, including the regularization parameter and kernel settings, should be selected using training data and validation procedures.

VIII. MODEL TRAINING AND SELECTION

Each candidate model should be trained using the same training dataset and a consistent evaluation protocol. Where practical, cross-validation can be used to estimate performance across multiple training and validation splits. The main steps are:

- Load the selected educational dataset.
- Define the target variable and eligible input features.
- Split the observations into training and testing subsets. 4) Fit preprocessing operations using training data.
- Train Logistic Regression, Decision Tree, Random Forest, and SVM models.
- Tune model hyperparameters using validation data or cross-validation.
- Select the model according to predefined evaluation criteria.
- Evaluate the selected model on the held-out test set.
- Record the results and document the experimental setup.

The final test set must not be repeatedly used to select models or adjust parameters. Repeatedly optimizing decisions against test outcomes can lead to an overly optimistic estimate of generalization performance.

IX. PERFORMANCE EVALUATION

Evaluation metrics depend on whether the task is classification or regression. Multiple metrics should be reported because different metrics reveal different aspects of predictive quality.

Classification Metrics

For binary classification, the confusion matrix contains true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

Accuracy measures the proportion of correct predictions:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}. \quad (8)$$

Precision measures the proportion of positive predictions that are correct:

$$\text{Precision} = \frac{TP}{TP + FP}. \quad (9)$$

Recall measures the proportion of actual positive observations that the model identifies:

$$\text{Recall} = \frac{TP}{TP + FN}.$$

© 2026 IJSRET

(10) The F1-score is the harmonic mean of precision and recall:

$$F1 = 2 \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (11)$$

For multiclass classification, precision, recall, and F1score can be calculated using macro, micro, or weighted averaging. The averaging method should be stated explicitly. Confusion matrices and per-class recall are useful when the dataset contains imbalanced performance categories.

Regression Metrics

If the target is a numerical examination score, regression metrics are more appropriate.

Mean Absolute Error (MAE) is

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \quad (12)$$

Root Mean Squared Error (RMSE) is

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \quad (13)$$

Here, y_i denotes the actual value and $\hat{y}_i$ denotes the predicted value. Lower MAE and RMSE generally indicate smaller prediction errors on the evaluated dataset.

The coefficient of determination, R^2, can also be reported for regression, with its interpretation depending on the data and evaluation conditions.

X. EXPERIMENTAL DESIGN AND RESULTS

The experimental evaluation should be conducted using a documented dataset and a reproducible implementation. The dataset name, number of observations, feature definitions, target construction, preprocessing steps, and software environment should be recorded.

The proposed framework supports experiments using the four classification algorithms described earlier. If numerical marks are predicted instead, appropriate regression algorithms and metrics should be included.

Experimental Configuration

Table II provides a configuration template. Values must be updated to match the actual implementation.

Results Reporting

The performance of each model should be measured on the same held-out test data. Table III is a reporting template and does not contain experimental results.

The dashes in Table III must be replaced with measured values after running the experiments. Results should be reported with consistent decimal precision and with the averaging method specified for each metric.

If the target is a numerical score, the comparison table should instead report MAE, RMSE, and, where appropriate, R2. Classification and regression metrics should not be compared as if they measured the same quantity.

Interpretation of Results

A model should not be selected solely because it produces the highest accuracy. In an imbalanced dataset, a model may obtain high accuracy while performing poorly on the minority class. Macro F1-score and per-class recall can help reveal such behavior.

The selected model should also be assessed for stability, interpretability, and computational requirements. Where feature importance is reported, the analysis should explain how importance was calculated and acknowledge that feature importance does not establish causation.

© 2026 IJSRET

No algorithm can be declared the best for the proposed application without actual experimental measurements. The final paper should therefore describe the observed results, explain differences between the models, and discuss possible sources of error based on the evidence.

XI. DISCUSSION

The proposed framework provides a structured approach for analyzing student academic records. Combining multiple features may help identify patterns that cannot be captured by a single academic indicator. Nevertheless, the usefulness of such a model depends on the quality and relevance of the available data.

Attendance, previous scores, and internal assessments may contain predictive information, but their relationships with academic outcomes can vary across institutions, courses, and student populations. Models should therefore be evaluated in the context where they are expected to be used.
A further consideration is the difference between prediction and intervention. Identifying a student as potentially at risk does not explain the underlying cause of the difficulty.

Educators should examine the student's circumstances and provide appropriate support rather than treating a model output as a definitive judgment.
A. Practical Applications
The framework may support several academic activities:

- Early identification: Flagging students who may benefit from additional academic attention.
- Academic advising: Supporting discussions between students, teachers, and advisors.
- Learning support: Helping institutions investigate where additional learning resources may be useful.
- Performance monitoring: Summarizing academic patterns across assessment periods.
- Institutional planning: Supporting data-informed reviews of teaching and learning processes.

These applications should be implemented with appropriate human oversight, access controls, and clear communication about the limits of predictions.

XII. LIMITATIONS AND ETHICAL CONSIDERATIONS

The proposed framework has several limitations. First, its predictive performance depends on the quantity, accuracy, and representativeness of the dataset. Missing or outdated records can reduce reliability. A model trained on one institution's students may not generalize to another institution without additional evaluation.

Second, correlations in historical educational data do not necessarily indicate causal relationships. A feature associated with low performance should not automatically be interpreted as the cause of that outcome.

Third, the model may reproduce biases present in historical records. Evaluation should examine whether prediction errors differ across relevant student groups, where such analysis is lawful, justified, and supported by sufficient data. Sensitive attributes should be handled with care.
Fourth, privacy and security are important because student records may contain confidential information. Data collection should be authorized, access should be restricted, and retention should follow applicable institutional policies and legal requirements. Finally, predicted performance categories can influence how students are treated. The system should not be used as the sole basis for denying educational opportunities, assigning punitive labels, or making consequential decisions. Predictions should be reviewed by qualified educators and used to offer support.

XIII. FUTURE ENHANCEMENTS

Several enhancements can extend the proposed framework. First, future work can evaluate the system using larger datasets collected across multiple academic terms and institutions. Such evaluation can help determine whether the learned patterns remain useful under different conditions.
Second, hyperparameter optimization and cross-validation can be used to investigate model configurations more

© 2026 IJSRET

systematically. Where appropriate, probability calibration can improve the interpretation of predicted probabilities.

Third, explainability techniques can be introduced to help educators understand which features contributed to individual predictions. These explanations should be validated and should not be treated as evidence of causation.

Fourth, a web-based interface could allow authorized users to submit records and view predictions, evaluation summaries, and appropriate explanations. Such an interface would require secure authentication, role-based access, input validation, and safeguards against exposing student information.

Finally, future studies can investigate model drift, fairness across groups, and the effect of academic interventions. Longitudinal evaluation is particularly important if the system is used to support decisions over multiple semesters.

XIV. CONCLUSION

This paper presented a machine-learning framework for student performance prediction using educational data. The proposed workflow includes data collection, preprocessing, feature preparation, model training, evaluation, and prediction. Logistic Regression, Decision Tree, Random Forest, and Support Vector Machine were identified as candidate classification algorithms for comparative evaluation.

The framework emphasizes consistent preprocessing, appropriate separation of training and testing data, and the use of multiple evaluation metrics. When the target is a numerical score, regression metrics should replace classification metrics. Actual performance must be established through experiments using a clearly documented dataset; no specific accuracy or model ranking can be concluded without those measurements.

Student performance prediction can provide useful decision-support information when implemented carefully. Its value lies in helping educators investigate learning needs and provide timely support, not in replacing professional judgment. Future work should prioritize external validation, explainability, fairness, privacy, and the evaluation of real educational interventions

REFERENCES

1. C.Romero and S. Ventura, “Educational data mining: A review of the state of the art,” IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 40, no. 6, pp. 601–618, Nov. 2010, doi: 10.1109/TSMCC.2010.2053532.
2. R. S. J. d. Baker and P. S. Inventado, “Educational data mining and learning analytics,” in Learning Analytics: From Research to Practice,J. A. Larusson and B. White, Eds. New York, NY, USA: Springer, 2014,pp. 61–75, doi: 10.1007/978-1-4614-3305-7_4.
   Siemens and P. Long, “Penetrating the fog: Analytics in learning and education,” EDUCAUSE Review, vol. 46, no. 5, pp. 30–40, 2011.
3. L. Breiman, “Random forests,” Machine Learning, vol. 45, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.
4. Cortes and V. Vapnik, “Support-vector networks,” Machine Learning, vol. 20, pp. 273–297, 1995, doi: 10.1007/BF00994018.
5. R. Quinlan, “Induction of decision trees,” Machine Learning, vol. 1,pp. 81–106, 1986, doi: 10.1007/BF00116251.
6. M. Bishop, Pattern Recognition and Machine Learning. New York, NY, USA: Springer, 2006.
7. Pedregosa et al., “Scikit-learn: Machine learning in Python,” Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011

© 2026 IJSRET

© 2026 IJSRET