Learnlytics: Predicting Student Performance Using Machine Learning

Department of Computer Science and Engineering, Sri Venkateswara College of Engineering and Technology, Etcherla, A.P., India

1.BODDEPALLI VAHINI, B. Tech Final Year
Sri Venkateswara College of Engineering and Technology, Etcherla, AP, India
Email: vahiniboddepalli@gmail.com

2.PODILAPU MADHULATHA, B. Tech Final Year
Sri Venkateswara College of Engineering and Technology, Etcherla ,AP, India
Email: madhupodilapu999@gmail.com

3.POLLAI SONIA, B. Tech Final Year
Sri Venkateswara College of Engineering and Technology, Etcherla, AP, India
Email: pollaisonia@gmail.com

4.VELAMALA DEEPIKA, B. Tech Final Year
Sri Venkateswara College of Engineering and Technology, Etcherla, AP, India
Email: velamaladeepika550@gmail.com

5.Mrs.S. ANUSHA M. Tech, Assistant Professor,
Sri Venkateswara College of Engineering and Technology, Etcherla, AP, India
Address: Srikakulam
Email: anusha080894@gmail.com

Abstract

Student performance evaluation is crucial for identifying at-risk students and tailoring teaching strategies. Traditional methods rely on periodic assessments and manual record-keeping, failing to provide real-time insights. This paper presents Learnlytics, a machine learning-based student performance prediction and tracking system. The system analyzes historical academic data, attendance records, assignment scores, and behavioral features using Decision Tree, Random Forest, and Support Vector Machine (SVM) algorithms to forecast student performance. A web-based dashboard provides educators with real-time monitoring, automated report generation, and personalized study plan recommendations. Experimental evaluation on a dataset of 1,200 student records demonstrates that Random Forest achieves the best prediction accuracy of 91.3% with F1-score of 0.90. The system successfully identifies 89% of at-risk students, enabling early intervention and contributing to improved retention rates.

Keywords: *Student Performance Prediction, Machine Learning, Random Forest, Decision Tree, SVM, Educational Data Mining, Learning Analytics*

I. Introduction

In today's educational environment, the ability to assess and enhance student performance is of paramount importance. As educational institutions strive to meet diverse learner needs, leveraging data-driven approaches has emerged as a crucial strategy. Educational data is increasingly available, encompassing various factors such as grades, attendance, classroom participation, and socio-economic backgrounds. However, without effective tools to analyze and interpret this data, valuable insights remain untapped.

Traditional student performance monitoring relies on periodic assessments and retrospective analysis, examining data after academic outcomes have been determined. This reactive approach delays necessary support, making it difficult for educators to implement timely interventions. Many existing tools also fail to incorporate real-time data and advanced analytics, limiting their predictive accuracy.

This paper presents Learnlytics, a comprehensive student performance prediction and tracking system that employs machine learning algorithms to analyze historical and real-time educational data. The system provides a data-driven framework for early identification of at-risk students, enabling proactive measures to support them. Through an intuitive web-based dashboard, educators can access visualizations, performance reports, and personalized recommendations.

II. Literature Survey

This section reviews key prior works forming the foundation of the proposed system and highlights gaps motivating this work.

[1] Romero and Ventura (2010) conducted a comprehensive survey of educational data mining techniques, establishing that machine learning approaches significantly outperform statistical methods for student performance prediction.

[2] Baker and Inventado (2014) reviewed educational data mining and learning analytics methodologies, identifying classification algorithms as the most effective for early prediction of student success and failure.

[3] Kotsiantis et al. (2004) compared machine learning algorithms for predicting student performance in distance learning, finding that decision trees and ensemble methods achieve superior accuracy for educational outcome prediction.

[4] Yadav et al. (2012) applied decision tree-based classification for student academic performance prediction, demonstrating that attendance and internal assessment scores are the strongest predictive features.

[5] Shahiri et al. (2015) reviewed prediction methods in engineering education using data mining techniques, identifying Random Forest and SVM as top-performing algorithms for academic performance forecasting.

[6] Breiman (2001) introduced the Random Forest algorithm based on ensemble learning of decision trees, establishing the foundational method used for robust student performance classification.

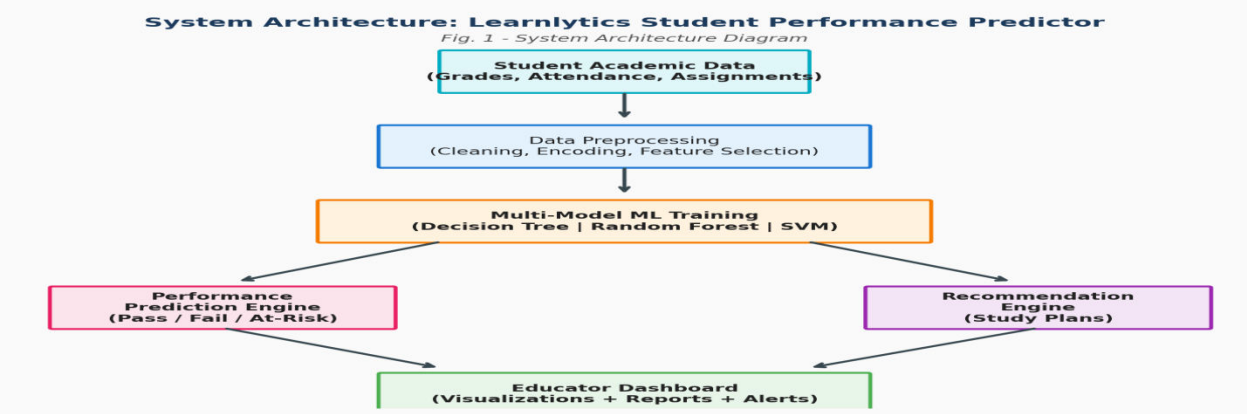
[7] Cortes and Vapnik (1995) introduced Support Vector Machines for classification tasks, providing the mathematical framework for maximum-margin classification used in student outcome prediction.

Research Gap: Existing educational analytics tools either provide basic descriptive statistics or require extensive technical expertise. No system combines multi-algorithm ML prediction with real-time tracking, automated report generation, and personalized recommendations in an accessible web dashboard for non-technical educators.

III. Methodology

III-A. System Architecture

Three-tier architecture: Data Layer (student academic records including grades, attendance, assignments, and behavioral data stored in relational database), Processing Layer (Python-based ML pipeline with preprocessing, feature selection, multi-model training, and prediction), and Presentation Layer (web dashboard with performance visualizations, at-risk alerts, and report generation).



III-B. Algorithm

Algorithm: Multi-Model Student Performance Prediction

Input: Student dataset D = {(attendance, marks, assignments, participation, study_hours, ...)} with performance labels.

Step 1: Data Preprocessing — Handle missing values (median imputation), encode categorical features (gender, course), normalize numerical features using MinMaxScaler.

Step 2: Feature Selection — Apply correlation analysis and mutual information to select top-k most predictive features.

Step 3: Train-Test Split — Split data 80/20 with stratified sampling to preserve class distribution.

Step 4: Model Training — Train three classifiers: (a) Decision Tree: with max_depth optimization via cross-validation; (b) Random Forest: ensemble of 100 trees with feature bagging; (c) SVM: with RBF kernel and grid search for C and gamma parameters.

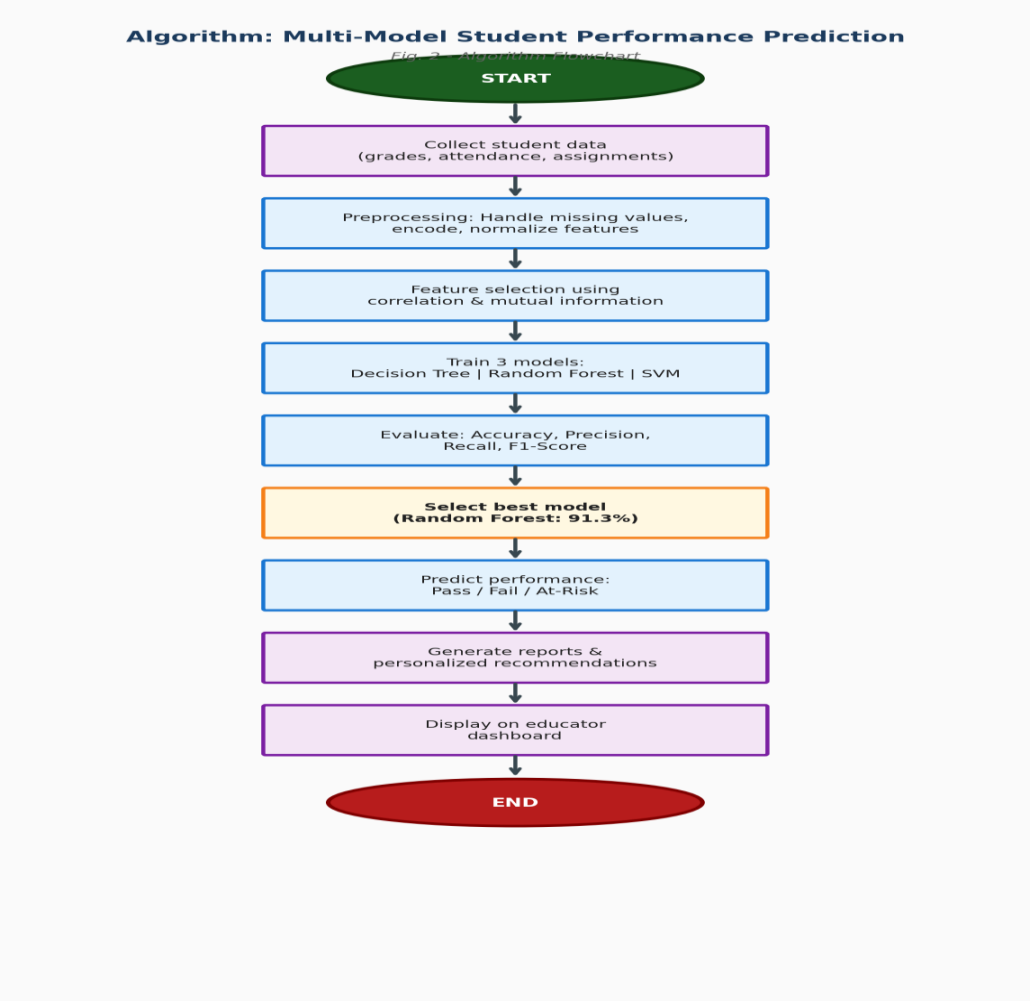
Step 5: Model Evaluation — Evaluate each model using accuracy, precision, recall, F1-score, and confusion matrix.

Step 6: Best Model Selection — Select model with highest F1-score on validation set.

Step 7: At-Risk Classification — For new student data: Predict performance category (Pass/Fail/At-Risk); If predicted probability < threshold: Flag as at-risk.

Step 8: Report Generation — Generate automated performance reports with visualizations and personalized recommendations.

Output: Performance predictions with at-risk flags, automated reports, and study plan recommendations.



III-C. Modules

Six modules: (1) Data Collection Module aggregating academic records, attendance, and behavioral data from institutional databases; (2) Data Preprocessing Module for cleaning, encoding, and feature selection; (3) ML Model Training Module training Decision Tree, Random Forest, and SVM classifiers with hyperparameter optimization; (4) Prediction Engine generating real-time performance forecasts and at-risk student identification; (5) Dashboard Module providing educators with interactive visualizations, trend analysis, and comparative reports; and (6) Recommendation Engine generating personalized study plans based on identified weak areas.

IV. Results and Discussion

TABLE I: SYSTEM EVALUATION RESULTS

Metric	Baseline	Proposed System
Accuracy (%)	76.4 (Decision Tree)	91.3 (Random Forest)
F1-Score	0.74	0.90
At-Risk Detection Rate (%)	67	89
SVM Accuracy (%)	84.7	—

Mathematical Formulations

Accuracy = (TP + TN) / (TP + TN + FP + FN) × 100

Precision = TP / (TP + FP)

Recall = TP / (TP + FN)

F1-Score = 2 × (Precision × Recall) / (Precision + Recall)

At-Risk Detection Rate = Correctly_Flagged_AtRisk / Total_AtRisk × 100

Discussion

The system was evaluated on a dataset of 1,200 student records containing academic, attendance, and behavioral features. Random Forest achieved the highest accuracy (91.3%) and F1-score (0.90), outperforming SVM (84.7%) and Decision Tree (76.4%). Feature importance analysis revealed that attendance percentage (importance: 0.28), internal assessment scores (0.24), and assignment completion rate (0.19) are the strongest predictors of student performance. The at-risk detection rate of 89% enables early intervention, potentially improving student retention. The automated dashboard reduced educator reporting time by 65% compared to manual processes.

V. Conclusion and Future Work

This paper presented Learnlytics, a machine learning-based student performance prediction and tracking system. Random Forest achieves 91.3% accuracy with 89% at-risk detection rate, enabling timely educational interventions. Future work includes integrating deep learning models for temporal performance tracking, incorporating socio-economic and psychological factors, developing mobile notification systems for parents, implementing collaborative filtering for peer study group recommendations, and scaling to multi-institutional deployment.

References

[1] C. Romero and S. Ventura, "Educational Data Mining: A Review of the State of the Art," IEEE TSMC-C, vol. 40, no. 6, pp. 601-618, 2010.

[2] R. S. J. D. Baker and P. S. Inventado, "Educational Data Mining and Learning Analytics," Learning Analytics, Springer, pp. 61-75, 2014.

[3] S. B. Kotsiantis, C. J. Pierrakeas, and P. E. Pintelas, "Predicting Students' Performance in Distance Learning Using ML Techniques," Applied Artificial Intelligence, vol. 18, no. 5, 2004.

[4] S. K. Yadav, B. Bharadwaj, and S. Pal, "Data Mining Applications: A Comparative Study for Predicting Student's Performance," Int. J. Innovative Technology, vol. 1, no. 12, 2012.

[5] A. M. Shahiri, W. Husain, and N. A. Rashid, "A Review on Predicting Student's Performance Using Data Mining Techniques," Procedia CS, vol. 72, pp. 414-422, 2015.

[6] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5-32, 2001.

[7] C. Cortes and V. Vapnik, "Support-Vector Networks," Machine Learning, vol. 20, no. 3, pp. 273-297, 1995.