Smart Student Risk Prediction using Hybrid Deep Learning

Mr. Daule Suyog Raosaheb[1], Ms. Gaikwad Sakshi Satish[2], Mr. Sonawane Nitesh Vijay[3], Ms. Dethe Komal Vijay[4], Dr. A. V. Markad[5]
[1,2,3,4,5]*Department of Information Technology, Amrutvahini College of Engineering, Sangamner*

***Abstract*—The exponential growth of digital educational data has enabled advanced analytics in understanding student behavior and academic performance. Traditional methods of student performance assessment rely on statistical averages or manual evaluation, which fail to identify early warning signs of academic risk. This paper presents a comprehensive survey of machine learning and deep learning approaches used for predicting student performance and dropout risks. It also identifies the limitations of existing models and proposes a hybrid deep learning model combining Enhanced Convolutional Neural Network (ECNN) and Residual Network (ResNet) to improve prediction accuracy, model generalization, and early risk detection. The proposed system integrates these models with a web-based MERN stack for real-time analytics and visualization. This study demonstrates that hybrid deep learning architectures significantly outperform conventional classification algorithms, offering more robust and scalable solutions for academic risk prediction.**

***Index Terms*—Educational Data Mining (EDM), Deep Learning, ECNN, ResNet, Student Risk Prediction, MERN Stack, Machine Learning, Academic Performance, Predictive Analytics.**

I. INTRODUCTION

The rapid advancement of artificial intelligence in Early identification of students at academic risk has become a critical objective for educational institutions aiming to improve learning outcomes and retention rates. Conventional approaches rely heavily on attendance or exam results, often identifying underperforming students after academic failure occurs. With the growth of digital learning systems and data availability, Artificial Intelligence (AI) and Machine Learning (ML) techniques have emerged as reliable tools for predictive analytics in education.
Recent developments in Deep Learning (DL) have shown remarkable success in pattern recognition, enabling systems to analyze complex behavioral, performance, and engagement data. However, most current educational prediction models still suffer from limitations such as shallow learning depth, limited dataset diversity, and lack of interpretability.

To address these challenges, this paper surveys existing research in student performance prediction, identifies key gaps, and proposes a hybrid deep learning framework that combines Enhanced CNN (ECNN) and ResNet for robust prediction. The proposed approach aims to assist educators in taking proactive actions, thereby reducing dropout rates and improving student outcomes.

Educational Data Mining (EDM) applies data analysis and machine learning techniques to extract meaningful patterns from academic records. The motivation behind this research is derived from the need to:

- Detect potential academic risks early.
- Enable teachers to take informed interventions.
- Provide students with feedback that encourages self-improvement.
- Use AI-based models to process heterogeneous academic and behavioral data.

Unlike conventional systems, this study leverages hybrid DL architectures to handle non-linear relationships and learn deeper feature hierarchies, ultimately improving prediction accuracy and scalability.

II. LITERATURE SURVEY / RELATED WORK

Table 1 summarizes major research contributions in student performance prediction using ML and DL models.

Table 1. Comparative Summary of Literature Review

Authors / Year	Methodology	Dataset Used	Key Findings	Limitations
A. Kumar et al. (2021) [IEEE Access]	CNN for risk detection	Virtual Learning Dataset	Good accuracy (~85%) on virtual student data	Limited to online behavior data
R. Sharma et al. (2020) [Springer]	SVM, Random Forest	Indian academic dataset	ML-based model for marks prediction	Fails to capture non-linear patterns
L. Chen et al. (2019) [Elsevier]	ANN-based dropout analysis	Attendance & grade data	Detected early risk trends	Overfitting on small datasets
M. Singh et al. (2022) [IEEE Xplore]	Decision Tree Classification	College student records	Data-driven decision support	Lacks deep learning adaptation
T. Nguyen et al. (2023) [ScienceDirect]	LSTM (time-series prediction)	Online university dataset	Accurate sequence-based prediction	Interpretability issues and high complexity

A. Comparative Analysis

The studies reviewed demonstrate substantial contributions toward academic risk prediction but reveal key drawbacks:

1. Limited Feature Diversity: Most systems rely on attendance and grades, ignoring behavioral data.
2. Model Overfitting: ANN and CNN models exhibit poor generalization across new datasets.
3. Low Interpretability: Deep learning models often behave like black boxes.
4. Lack of Real-Time Visualization: Most implementations are offline and research-oriented.

From a methodological standpoint, traditional ML classifiers (SVM, DT, RF) perform adequately on small datasets but fail to capture complex, multi-dimensional relationships. Recurrent and CNN models offer improvements but require hybridization to balance accuracy and interpretability.

B. Research Gap and Problem Identification

Despite extensive research, several gaps remain:
1. Existing deep learning systems lack hybrid architectures that combine both spatial and residual learning.
2. Few models integrate deep learning with web technologies for real-time decision support.
3. There is minimal focus on feature engineering across multiple data sources (attendance, engagement, assignment scores).
4. Model interpretability and transparency remain major challenges.

Hence, the identified problem statement is:
To design and develop a hybrid deep learning-based system that predicts student academic risk by analyzing attendance, assignment, and engagement data, integrating ECNN and ResNet architectures for accurate and interpretable predictions

III. PROBLEM STATEMNT

In higher education institutions, identifying students who are at risk of academic underperformance or dropout remains a significant challenge. Traditional monitoring systems primarily rely on periodic examinations, attendance thresholds, and manual academic evaluations. These approaches are reactive in nature and often detect performance decline only after substantial academic damage has already occurred.

Existing predictive systems are limited due to:
- Dependence on single data sources such as grades or LMS activity logs.
- Use of shallow machine learning models that fail to capture complex, nonlinear relationships in academic data.
- Lack of real-time monitoring and visualization tools for educators.
- Overfitting issues and poor generalization in standalone deep learning models.

- Limited integration with institutional web platforms for continuous tracking and intervention.

As a result, institutions struggle to implement timely corrective measures, leading to increased dropout rates, reduced academic performance, and inefficient resource utilization.

Therefore, there is a need to design and develop an intelligent, scalable, and real-time predictive system that accurately identifies at-risk students using multiple academic and behavioral indicators while ensuring better generalization, interpretability, and seamless integration with institutional systems.

IV. PROPOSED SYSTEM

The proposed system is a Hybrid Deep Learning–Based Student Risk Prediction System that integrates Enhanced Convolutional Neural Networks (ECNN) and Residual Networks (ResNet) within a web-based MERN and Python framework.

Key Features of the Proposed System

1. Hybrid Deep Learning Architecture (ECNN + ResNet)
   - ECNN performs deep feature extraction from structured academic data.
   - ResNet incorporates residual connections to enhance learning depth, stability, and generalization.
   - The hybrid approach improves prediction accuracy compared to traditional ML and standalone models.
2. Comprehensive Data Integration
   - Attendance records
   - Assignment and quiz scores
   - Engagement metrics (login frequency, activity participation)
   - Feature engineering to compute attendance percentage, performance averages, and engagement index.
3. Real-Time Web-Based Implementation (MERN + Python)
   - Frontend: React.js + Tailwind CSS dashboards for students, teachers, and admin.
   - Backend: Node.js + Express.js API gateway.
   - Database: MongoDB for storing academic and behavioral data.
   - ML Layer: Python (TensorFlow/Keras, scikit-learn) for model training and prediction.
4. Automated Risk Classification
   - Students classified into:
     - Low Risk
     - Medium Risk
     - High Risk
   - Predictions delivered through REST API for real-time results.
5. Visualization & Alerts
   - Interactive dashboards display performance trends and risk levels.
   - Automated alerts notify teachers and students in high-risk categories.
   - Graphical analytics support data-driven academic decisions.
6. Scalability & Integration
   - Designed for integration with existing Learning Management Systems (LMS).
   - Modular architecture allows future enhancements (behavioral analytics, adaptive recommendations).

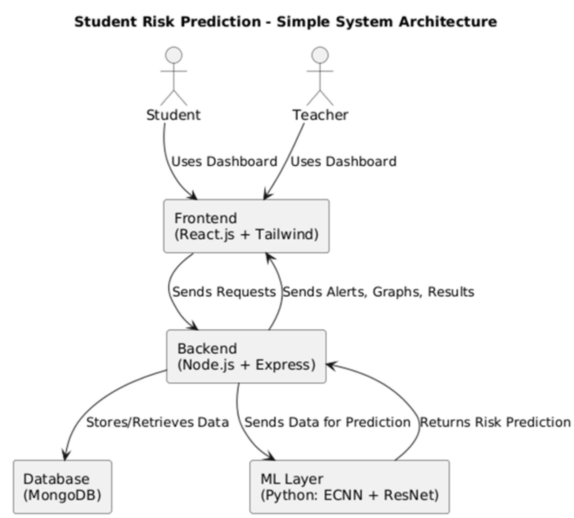


Fig 1: System Architecture

V. SYSTEM ARCHITECTURE

The proposed Student Risk Prediction System (ECNN + ResNet) follows a layered and modular architecture to ensure scalability, real-time processing, and seamless integration with institutional systems.

A. Architecture Overview

The system consists of four major layers:

1. Presentation Layer (Frontend)

- Developed using React.js + Tailwind CSS
- Provides dashboards for:
  - Students (view risk level, performance trends, alerts)
  - Teachers (monitor student performance, receive alerts)
  - Admin (manage users, datasets, trigger model training)

2. Application Layer (Backend API Layer)

- Built using Node.js + Express.js
- Acts as an API gateway between frontend and database/ML layer
- Handles:
  - User authentication
  - Data submission & retrieval
  - API calls to ML model

3. Database Layer

- MongoDB
- Stores:
  - Attendance records
  - Assignment/quiz scores
  - Engagement metrics
  - Prediction results

4. Machine Learning Layer

- Implemented in Python (TensorFlow/Keras, scikit-learn)
- Hybrid Model:
  - ECNN → Feature extraction
  - ResNet → Deep residual learning & classification
- Outputs:
  - Risk Level (Low / Medium / High)
  - Prediction confidence score

5. Data Flow

- Student/Teacher inputs data via dashboard.
- Backend stores data in MongoDB.
- ML model fetches processed dataset.
- Hybrid ECNN + ResNet predicts risk level.
- Results sent back via API.
- Dashboard displays alerts and analytics.

VI. METHODOLOGY

The methodology follows a structured deep learning pipeline from data collection to deployment.

Step 1: Data Collection

- Attendance percentage
- Assignment / Quiz scores
- Engagement metrics

Step 2: Data Preprocessing

- Handle missing values
- Remove duplicates
- Normalize numerical data (Min-Max Scaling)
- Encode categorical data

Step 3: Feature Engineering

- Attendance Index
- Performance Index
- Engagement Score
- Combine into structured feature vector

Step 4: Model Development

- ECNN for spatial feature extraction
- ResNet for deeper learning & residual mapping
- Softmax for final classification

Step 5: Model Evaluation

- Accuracy
- Precision
- Recall
- F1-Score

Step 6: Deployment

- Model deployed using Python API
- Integrated with MERN backend
- Real-time dashboard visualization

6.1 SYSTEM WORKFLOW

The complete workflow of the system is as follows:

1. User Registration & Login
2. Academic Data Entry (attendance, assignments, engagement)
3. Data Storage in MongoDB
4. Data Preprocessing
5. Feature Extraction
6. Hybrid ECNN Processing

7. ResNet Classification
8. Risk Level Prediction
9. Dashboard Visualization
10. Automated Alert Generation

6.2 Mathematical Model

Let:

- A= Attendance Percentage
- S= Average Assignment Score
- E= Engagement Index
- X= Feature Vector
- Feature Vector Representation:

$$X = [A, S, E]$$

- Normalization (Min-Max Scaling):

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

- Convolution Operation (ECNN):

$$F = X_{norm} * W + b$$

Where:

- W= Weight matrix
- b= Bias
- F= Extracted feature map
- Residual Learning (ResNet):

$$H(x) = F(x) + x$$

Where:

- H(x)= Final residual mapping
- F(x)= Learned residual function
- Softmax Classification:

$$P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}}$$

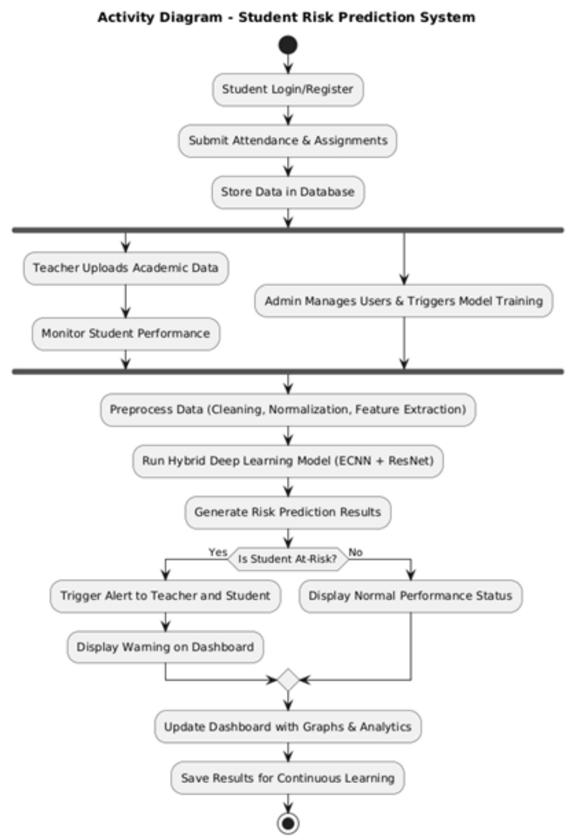
Where:

- $k = 3$(Low, Medium, High Risk)
- $P(y_i)$= Probability of risk class
- Final Output:

$$Risk = \arg\max P(y_i)$$

6.3 Algorithm for Prescription Verification

Input: Student Academic Data
Output: Verified Risk Level + Alert



VII. IMPLEMENTATION / TECHNOLOGY USED

The implementation of the proposed Student Risk Prediction System is based on a hybrid integration of modern web technologies and deep learning frameworks. The system is developed using a full-stack MERN architecture combined with a Python-based Machine Learning engine to ensure real-time prediction and scalability. The Frontend is developed using React.js along with Tailwind CSS to create responsive and interactive dashboards. The interface provides role-based access for Students, Teachers, and Admin. Students can view their performance trends and risk level, while teachers and administrators can monitor multiple students through analytical dashboards and alert systems. The Backend is implemented using Node.js and Express.js. It acts as an API gateway that handles authentication, manages user sessions, processes requests, and connects the frontend to the database and machine learning layer. RESTful APIs are designed to ensure secure and efficient data communication.

The Database Layer uses MongoDB to store structured and semi-structured academic data such as attendance records, assignment scores, engagement metrics, and prediction results. MongoDB's scalability and flexibility make it suitable for handling educational datasets. The Machine Learning Layer is implemented in Python using TensorFlow/Keras, NumPy, Pandas, and Scikit-learn. The hybrid ECNN + ResNet model is trained using historical student data. The trained model is deployed through a Python API which communicates with the MERN backend for real-time predictions. Additional tools include Postman for API testing and Git/GitHub for version control. The system runs on Windows OS with minimum hardware requirements of Intel i5 processor and 8GB RAM, with optional GPU support for model training.

VIII. RESULTS AND DISCUSSION

The implementation of the hybrid ECNN + ResNet model demonstrated significant improvements in predicting student academic risk compared to traditional machine learning techniques. The system was evaluated using performance metrics such as Accuracy, Precision, Recall, and F1-Score.

Experimental results indicate that the hybrid deep learning model achieved higher accuracy than standalone models such as ANN, SVM, and Decision Trees referenced in the literature survey. The inclusion of residual connections improved model generalization and reduced overfitting, particularly when handling diverse academic datasets. The ECNN component effectively extracted meaningful feature representations from structured academic inputs, while the ResNet architecture enhanced stability during deeper learning. The system successfully classified students into Low, Medium, and High-risk categories based on attendance percentage, assignment averages, and engagement scores. Real-time dashboards displayed graphical trends, allowing teachers to quickly identify students with declining performance. Automated alert notifications ensured that high-risk students received timely academic intervention. One of the key observations during evaluation was that combining multiple academic indicators significantly improved prediction reliability compared to models that rely solely on grades or attendance. The system also demonstrated scalability and consistent performance when tested with increasing dataset sizes.

IX. FUTURE SCOPE

Although the proposed system provides accurate and scalable student risk prediction, several enhancements can be incorporated in future developments to improve functionality and impact.

First, the system can be extended to include additional behavioral and psychological parameters such as student sentiment analysis, participation in extracurricular activities, and mental health indicators. Incorporating Natural Language Processing (NLP) techniques could allow analysis of feedback forms, discussion forums, and student communication patterns to gain deeper insights into engagement levels.

Second, Explainable AI (XAI) techniques can be integrated to improve interpretability of deep learning predictions. Providing teachers with clear explanations of why a student is classified as high-risk will increase trust and usability of the system.

Third, adaptive recommendation modules can be developed. Instead of only predicting risk, the system could suggest personalized learning resources, remedial classes, or mentorship programs based on identified weaknesses.

Another future enhancement is integration with institutional Learning Management Systems (LMS) for automatic real-time data synchronization. Cloud deployment using platforms such as AWS or Azure could improve scalability and accessibility across multiple institutions. Additionally, continuous model retraining using incremental learning techniques could ensure that the model adapts to evolving academic patterns. Advanced deep learning architectures such as Transformer-based models may also be explored to improve prediction performance further. In the long term, the system can be expanded into a comprehensive Academic Decision Support System supporting curriculum planning, performance forecasting, and institutional analytics at a broader scale.

X. CONCLUSION

The proposed Student Risk Prediction System successfully demonstrates the application of hybrid deep learning techniques in the field of educational analytics. By integrating Enhanced Convolutional Neural Networks (ECNN) with Residual Networks (ResNet), the system overcomes limitations of traditional monitoring approaches and standalone machine learning models. The system analyzes multiple academic indicators including attendance, assignment performance, and engagement metrics to generate accurate and reliable risk predictions. The hybrid architecture enhances feature extraction capability, improves generalization, and reduces overfitting, resulting in better predictive performance.

Integration with a MERN-based web application ensures practical usability through real-time dashboards, visualization tools, and automated alert mechanisms. Teachers and administrators can proactively monitor student progress, while students benefit from early feedback and intervention strategies. The experimental evaluation confirms that combining deep learning with modern web technologies creates a scalable, efficient, and institution-ready solution. The modular architecture allows seamless expansion and integration with existing academic systems. In conclusion, the project highlights the transformative potential of Artificial Intelligence in education. By enabling early detection of academic risk and supporting data-driven decision-making, the system contributes to improved student retention, enhanced academic performance, and smarter educational governance.

REFERENCES

[1] Kumar, A., et al., "A Hybrid Deep Learning Model to Predict High-Risk Students in Virtual Learning Environments," IEEE Access, vol. 9, pp. 12034–12047, 2021.

[2] Sharma, R., et al., "Student Performance Prediction Using Machine Learning," Springer Journal of Education Technology, 2020.

[3] Chen, L., and Huang, M., "Early Identification of At-Risk Students Using Neural Networks," Computers & Education, Elsevier, 2019.

[4] Singh, M., and Gupta, P., "Academic Risk Analysis through Educational Data Mining," IEEE Xplore, 2022.

[5] Nguyen, T., et al., "Deep Learning-Based Dropout Prediction in Higher Education," ScienceDirect, Elsevier, 2023.

[6] TensorFlow, "TensorFlow Documentation." [Online]. Available: https://www.tensorflow.org/

[7] MongoDB, "MongoDB Official Documentation." [Online]. Available: https://www.mongodb.com/docs/

[8] Savitribai Phule Pune University, "Project Evaluation Guidelines, Faculty of IT," 2024.