CAPSTONE PROJECT

STUDENT PERFORMANCE PREDICTION USING MACHINE LEARNING

PRESENTED BY

STUDENT NAME: PAL SATYAM G

COLLEGE NAME: ADITYA SILVER OAK INSTITUTE OF TECHNOLOGY

DEPARTMENT: B.TECH (INFORMATION TECHNOLOGY)

EMAIL ID: satyampal200431@gmail.com

AICTE STUDENT ID: STU666efaad6524b1718549165

OUTLINE

- Problem Statement
- Proposed System/Solution
- System Development Approach
- Algorithm & Deployment
- Result (Output Image)
- Conclusion
- Future Scope
- References

PROBLEM STATEMENT

- Educational institutions collect a large amount of student data, but they often fail to use it effectively to identify students who may underperform.
- Key indicators such as classroom participation (e.g., raised hands, resource usage), discussions, and absenteeism are not properly analyzed in traditional systems.
- As a result, teachers lack tools to predict which students need academic support early.
- This project aims to solve this problem by applying machine learning algorithms to **predict student performance** — categorized as **Low (L), Medium (M), or High (H)** — based on behavioral and academic data from a classroom setting.

PROPOSED SOLUTION

- ❑ The proposed system is a **machine learning-based prediction model** that analyzes student classroom behavior and participation to classify their academic performance into three levels:
  High (H), Medium (M), and Low (L).
- This system is trained using a dataset containing various student attributes such as:
- Number of times they raised hands
- Number of resources accessed
- Participation in discussions
- Number of absence days
- Background details (grade, section, gender, nationality, etc.)

- ❑ The system follows these steps:

1. **Data Preprocessing** – Remove irrelevant columns and encode categorical data
2. **Model Training** – Train 5 different machine learning models
3. **Model Evaluation** – Compare models based on accuracy and classification reports
4. **Prediction** – Predict performance class (L, M, H) for new inputs.

- The system helps educators take early action for students likely to underperform.

SYSTEM APPROACH

❑ **Technology Stack:**

- Programming Language: Python
- Development Environment: Google Colab and Github

❑ **Libraries Used:**

- Pandas and NumPy for data handling
- Matplotlib and Seaborn for visualization
- Scikit-learn for machine learning models and evaluation

❑ **Dataset Overview:**

- File Name: AI-Data.csv
- Contains academic and behavioral records of students from different backgrounds
- Target Output: Class (L for Low, M for Medium, H for High)

❑ **Features Used:**

- Gender, Nationality, Stage, Grade, Section
- Raised Hands, Visited Resources, Announcements Viewed, Discussions Participated
- Student Absence Days

SYSTEM APPROACH

❑ **Preprocessing Steps:**

- Removed unnecessary columns like Place of Birth, Relation, Parent Satisfaction
- Converted text-based data into numeric format using label encoding

❑ **Selected main features:**

- Raised Hands, Visited Resources, Discussions, Absences

❑ **Train-Test Split:**

- Data divided into 70 percent training and 30 percent testing
- Training data used to build models
- Testing data used to evaluate accuracy

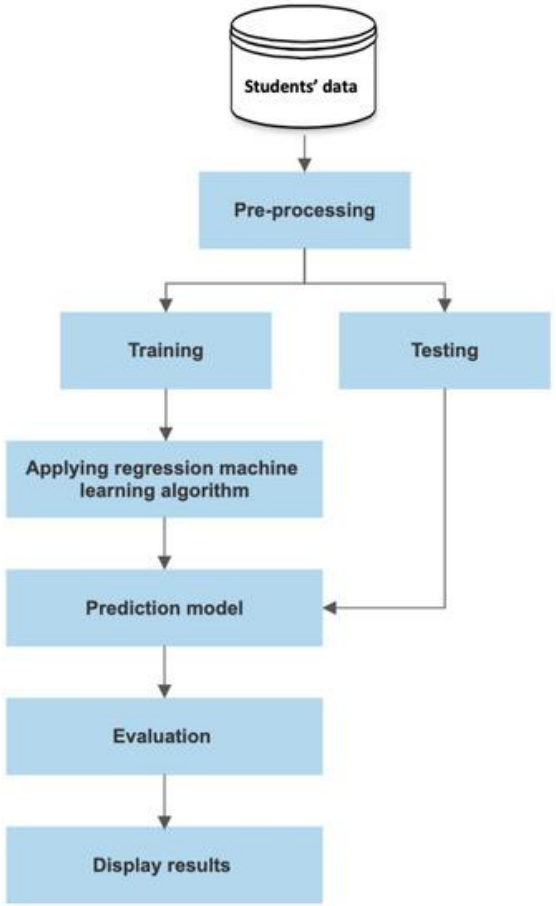

Figure 1. *The flowchart of the prediction model.*

ALGORITHM & DEPLOYMENT

❑ **Algorithms Implemented**

- Decision Tree Classifier
- Random Forest Classifier
- Perceptron (Linear Model)
- Logistic Regression
- Multi-Layer Perceptron (Neural Network)

❑ **Training Process**

- Data was loaded from AI-Data.csv
- Unnecessary columns were dropped such as gender, Stage ID, Grade ID, Nationality, Section, etc.
- Remaining categorical columns were label encoded into numerical form
- Selected input features:
- Raised Hands
- Visited Resources
- Discussion Participation
- Student Absence Days

ALGORITHM & DEPLOYMENT

❑ **The dataset was shuffled and then split into:**

- 70% for training
- 30% for testing
- Each model was trained using the training features and labels

❑ **Prediction and Evaluation:**

- Each model predicted the student performance class on test data
- Prediction results were compared with actual values
- Accuracy was calculated manually by counting correct predictions
- Classification reports were printed showing precision, recall, and F1-score
- Random Forest achieved the highest accuracy

❑ **Deployment Functionality:**

- The program allows the user to manually enter input values like raised hands, resource usage, etc.
- All five models return a predicted class: Low, Medium, or High
- The system prints model-wise predictions for comparison and decision support

RESULT

Accuracy measures using Decision Tree:

	precision	recall	f1-score	support
0	0.60	0.65	0.62	52
1	0.88	0.82	0.85	28
2	0.62	0.59	0.60	63
accuracy			0.66	143
macro avg	0.70	0.69	0.69	143
weighted avg	0.66	0.66	0.66	143

Accuracy using Decision Tree: 0.657

Accuracy Measures for Random Forest Classifier:

	precision	recall	f1-score	support
0	0.66	0.73	0.69	52
1	0.93	0.93	0.93	28
2	0.72	0.65	0.68	63
accuracy			0.73	143
macro avg	0.77	0.77	0.77	143
weighted avg	0.74	0.73	0.73	143

Accuracy using Random Forest: 0.734

Figure 2. *Accuracy measures using Decision tree and Random Forest Classifier*

Accuracy measures using Linear Model Perceptron:

	precision	recall	f1-score	support
0	0.40	1.00	0.57	52
1	1.00	0.14	0.25	28
2	0.00	0.00	0.00	63
accuracy			0.39	143
macro avg	0.47	0.38	0.27	143
weighted avg	0.34	0.39	0.26	143

Accuracy using Linear Model Perceptron: 0.392

Accuracy measures using Linear Model Logistic Regression:

	precision	recall	f1-score	support
0	0.67	0.85	0.75	52
1	0.93	0.89	0.91	28
2	0.78	0.62	0.69	63
accuracy			0.76	143
macro avg	0.79	0.79	0.78	143
weighted avg	0.77	0.76	0.75	143

Accuracy using Linear Model Logistic Regression: 0.392

Figure 3. *Accuracy measures using Linear Model Perception & Logistic Regression*

RESULT

```
Accuracy measures using MLP Classifier:
              precision    recall  f1-score   support

           0       0.64      0.83      0.72        52
           1       0.89      0.86      0.87        28
           2       0.73      0.57      0.64        63

    accuracy                           0.72       143
   macro avg       0.76      0.75      0.75       143
weighted avg       0.73      0.72      0.72       143


Accuracy using Neural Network MLP Classifier:  0.72
```

Figure 4. *Accuracy measures using MLP Classifier*

```
1.Marks Class Count Graph       2.Marks Class Semester-wise Graph
3.Marks Class Gender-wise Graph 4.Marks Class Nationality-wise Graph
5.Marks Class Grade-wise Graph  6.Marks Class Section-wise Graph
7.Marks Class Topic-wise Graph  8.Marks Class Stage-wise Graph
9.Marks Class Absent Days-wise  10.No Graph

Enter Choice:
```

Figure 5. *User input interface for generating student performance graphs.*

Sr. No.	Accuracy measures model	Accuracy
1	Decision Tree	0.675
2	Random Forest Classifier	0.734
3	Linear Model Perception	0.392
4	Logistic Regression	0.392
5	MLP Classifier	0.72

Table 1. *Accuracy measures result*

Note: *All models were trained, tested, and evaluated on Google Colab*.

RESULT

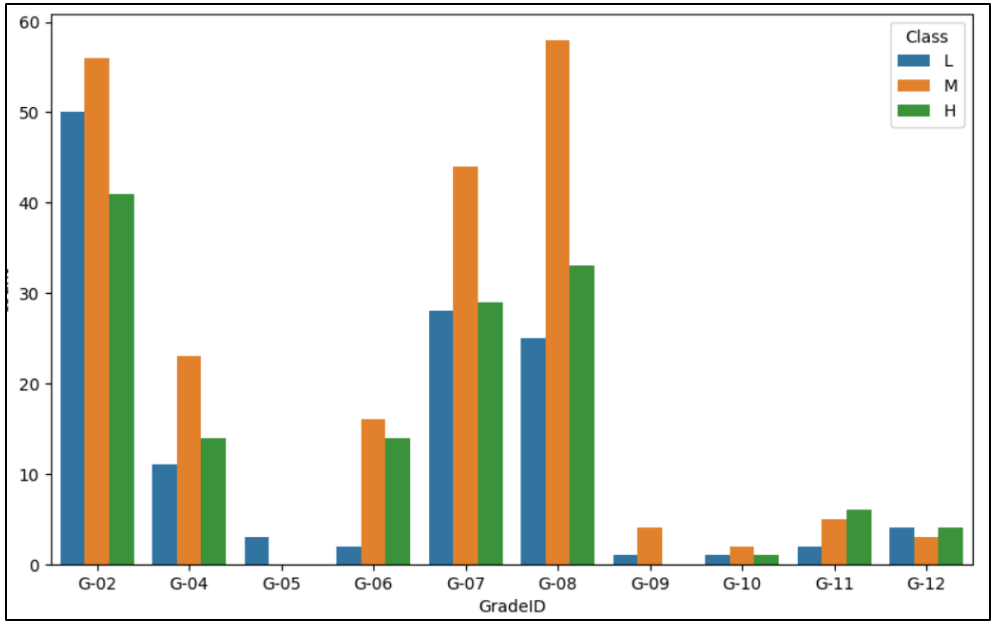

Figure 6. *Marks class Grade wise Graph*

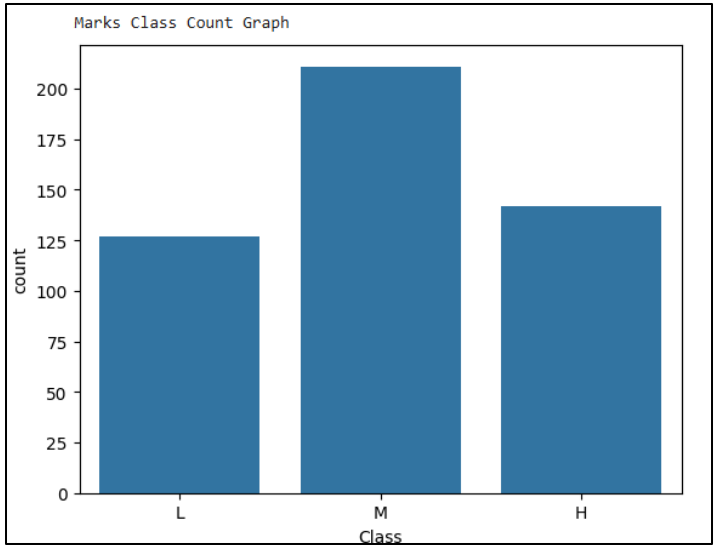

Figure 7. *Marks class Count Graph*

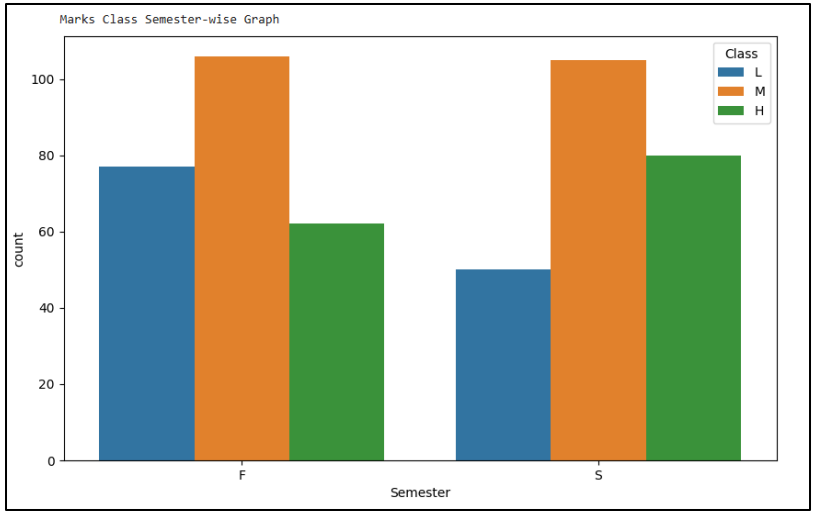

Figure 8. *Marks class Semester wise Graph*

- These visualizations show the distribution of students across different performance classes (Low, Medium, High) based on grade, total count, and semester.
- The graphs help identify patterns and trends useful for early academic intervention.
- These visuals help teacher to understand student performance.

CONCLUSION

- The project aimed to develop a machine learning-based system to predict student performance levels categorized as Low, Medium, or High. The analysis was performed using a real-world dataset consisting of various academic and behavioral features such as raised hands, visited resources, discussions participated, and student absence days.
- After cleaning and preprocessing the dataset, five machine learning algorithms were implemented: **Decision Tree, Random Forest, Perceptron, Logistic Regression**, and **MLP Classifier**.
- Each model was trained on 70% of the dataset and evaluated on the remaining 30%. Evaluation metrics such as accuracy, precision, recall, and F1-score were used to compare model performance. The **Random Forest Classifier** achieved the highest accuracy, followed by the **MLP Classifier**, indicating strong performance in classification tasks with imbalanced class distributions.
- The system also includes a user-input option to predict the class of a new student based on real-time input features. This functionality demonstrates the practical applicability of the model in academic settings.
- Overall, the system can help educational institutions identify at-risk students early and support data-driven decision-making to improve academic performance and learning outcomes.

FUTURE SCOPE

- This project lays the foundation for predictive analytics in the education sector. Several future improvements and enhancements can be made:
- **Feature Expansion:**
  Include more attributes such as exam scores, assignment submissions, attendance logs, and psychological factors (motivation, stress levels) to improve prediction accuracy.
- **Real-Time Integration:**
  Deploy the system with real-time data collection from smart classrooms, learning management systems, or school ERP software.
- **Improved Accuracy with Deep Learning:**
  Implement deep learning models like LSTM or CNN to learn complex patterns and improve performance for larger datasets.
- **Interactive Dashboard:**
  Develop a web-based dashboard where teachers and administrators can view predictions, trends, and at-risk students in real-time.
- **Early Intervention System:**
  Integrate with counseling or mentoring programs to automatically notify educators when a student is predicted to underperform.

REFERENCES

- Github Link **(STUDENT PERFORMANCE PREDICTION USING MACHINE LEARNING) :** https://github.com/Satyam-pal31/StudentPerformancePredictionUsingMachineLearning.git

- Student Performance Analysis and Prediction(Research Paper): https://www.analyticsvidhya.com/blog/2023/04/student-performance-analysis-and-prediction/

- Machine Learning Project on Student grade Prediction | Great Learning(Video): https://www.youtube.com/live/Y3y8v9PrMMY?si=mK_BgvCvgGvFN-53

- A Systematic Literature Review of Student' Performance Prediction using ML: https://www.mdpi.com/2227-7102/11/9/552

Thank you