# Content


1. Abstract
2. Introduction
3. Literature Review
4. Methodology
    - 4.1 Data Collection and Dataset Overview
    - 4.2 Data Preprocessing
    - 4.3 Model Selection and Training
    - 4.4 Evaluation Metrics
    - 4.5 Expandable AI (LIME & SHAP)
    - 4.6 Introduce a Web-Application
5. Results
    - 5.1 Data Preprocessing Result
    - 5.2 Performance Metrics
    - 5.3 ROC AUC Analysis
    - 5.4 Confusion Matrix Analysis
    - 5.5 Expandable AI (LIME) Result
        - 5.5.1 LIME Result
        - 5.5.2 SHAP Result
    - 5.6 The Web-Application
6. Conclusion and Future Work
7. List of Tables and Figures
8. References

# 1. Abstract

In recent years, machine learning (ML) techniques to predict and evaluate student performance has gained a strong significant attention in the field of education. This cloud be for higher studies or for kinder garden. It has been recognized by many researchers that students struggle with their academic when they face a curricula transfer. This study aims to explore the potential of machine learning in predicting student levels and assessing academic outcomes in next higher grade/year. The dataset “Student Level Prediction in UAE” [1], analysing a range of student data, including demographic, behavioural, and academic performance indicators, various ML models such as LightGBM, XGBoost and Voting Classifier (including: Logistic Regression, ANN, SVM, lightGBM) are employed to predict student success and identify at-risk learners early. The study highlights the advantages of using data-driven approaches in educational institutions to enhance personalized learning experiences, improve retention rates, and inform targeted interventions. Results from extensive experiments demonstrate the voting classifier achieve 92.9% of accuracy. Furthermore, the study emphasizes the importance of integrating machine learning models into the educational framework, as they offer valuable insights for curriculum design, teaching methodologies, and student support services. In addition, this study also focusses on Expandable AI, includes LIME (Local Interpretable Model-agnostic Explanations) and SHAP (SHapley Additive exPlanations) as well. The LightGBM and Voting Classifier models are used to study the impact of XAI and understand how these models predict the value, and which features effect the result and how much.

Ultimately, this research contributes to the growing body of literature on the application of machine learning in education, providing a foundation for future research aimed at further enhancing academic performance prediction and student evaluation systems.

# 2. Introduction

The rapid advancements in technology over the past decade have revolutionized various fields, and education is no exception. As educational institutions continue to evolve and expand, there is a growing need for innovative approaches to support and enhance student learning. Traditionally, academic success has been evaluated based on grades, test scores, and teacher assessments. However, these methods fail to capture the full spectrum of factors influencing a student’s academic journey, such as behavioural patterns, personal circumstances, previous and current curricula and engagement with learning resources. This has led to a growing interest in leveraging data and machine learning (ML) algorithms to predict student performance and develop more personalized, data-driven educational strategies.

Machine learning, a subset of artificial intelligence, has proven to be highly effective in extracting patterns from large datasets, making it an ideal tool for analysing student data. The ability to predict academic outcomes before they manifest can enable educators to implement timely interventions, personalize learning experiences, and identify at-risk students long before they fail or disengage. By utilizing data from various sources, including student demographics, attendance records, participation in online learning environments, and prior academic performance, machine learning algorithms can provide valuable insights into student success and failure factors. These insights are crucial for educational institutions

aiming to improve retention rates, optimize teaching methods, and tailor curriculum designs to better meet student needs.

Despite the promise of machine learning in education, many challenges remain in its application to student performance prediction. Traditional machine learning models such as Decision Trees, Support Vector Machines (SVM), and Neural Networks are frequently used for predicting academic outcomes. However, these models often struggle with issues such as overfitting, lack of interpretability, and inability to handle diverse datasets with complex relationships between features. As a result, many studies in the field of student performance prediction have highlighted the need for more robust and accurate models that can overcome these limitations.

Recent developments in ensemble learning methods have addressed some of these challenges. Ensemble methods, which combine multiple models to improve predictive performance, have demonstrated significant potential in various domains, including student performance prediction. By integrating the strengths of multiple machine learning algorithms, ensemble models can produce more stable, accurate, and interpretable results. Moreover, the application of graph-based methods in combination with ensemble learning offers an exciting avenue for improving prediction accuracy. Graph-based ensemble methods allow for the propagation of information through connected data points, creating a more coherent and reliable prediction model that can better reflect the dynamic nature of student performance.

This study aims to explore the potential of using machine learning, particularly ensemble and graph-based methods, for predicting and evaluating student performance. By developing and evaluating multiple ML models on diverse student data, this research seeks to determine the most effective approach for predicting student success, identifying at-risk students, and supporting tailored interventions. The results of this research will provide a deeper understanding of how data-driven approaches can revolutionize student evaluation, offer practical applications for educators, and contribute to the growing body of knowledge on the role of artificial intelligence in education. Not just focused on the outcomes of each model, but this study also insight about the expandable AI XAI and give a brief introduction to LIME technology. (Local Interpretable Model-agnostic Explanations), and how it works to explain the prediction of machine learning models.

The remainder of this paper is organized as follows: Section 2 reviews the existing literature on machine learning applications in student performance prediction, highlighting the strengths and limitations of current approaches. Section 3 describes the research methodology, including the datasets used, the machine learning models implemented, and the evaluation metrics employed. Section 4 presents the results of the experiments and discusses the findings. Finally, Section 5 concludes with recommendations for future research and the practical implications of the study’s findings for educational institutions.

In addition to the machine learning models developed for student performance prediction, this study also introduces a web application designed to provide an interactive and user-friendly interface for students and educators. The web application allows users to input various student-related data, such as demographics, academic performance, and curriculum details, to receive predictions regarding the student's future academic success. By integrating the machine learning model into the web app, this tool aims to bring data-driven insights directly to students, helping them understand their academic trajectory and potential areas for improvement. This integration not only enhances the practical application of the study but also empowers students and educators with actionable insights to support personalized learning strategies and timely interventions.

# 3. Literature Review

The use of machine learning (ML) to predict student performance has grown rapidly in recent years, thanks to the large amount of educational data being generated by institutions. As education systems continue to develop, it's becoming more important to create methods that not only predict academic success but also identify students at risk early and offer the necessary support.

In diverse environments, where multiple education curriculums are in place, machine learning is being used to tackle challenges in placing students at the right academic level. Ghareeb et al. [2] suggest a framework that uses ML algorithms to help assign students to the appropriate year group. This is especially important as curriculums can differ greatly in areas like assessment methods, exam boards, and academic calendars. By using machine learning classifiers like Random Forest and Artificial Neural Networks, their approach makes it easier for students to transition between curriculums, ensuring smoother integration and better monitoring of their academic progress.

Another journal by Shilpa M. et al., [3] however, India's traditional teaching methods make it difficult to track student progress. The lack of standard assessment practices and a vast student population further complicates performance monitoring. This study explores factors like age, health, and parents' background, using visualisation to identify weak students early. Machine learning models, KNN, Logistic Regression, and SVM were applied. The SVM model with a linear kernel achieved the best accuracy of 84.37%, making it the most effective.

Predicting how well students will perform academically has been a growing concern in education, and researchers are finding more ways to use data and technology to help with this. One such approach is Educational Data Mining (EDM), which looks at various factors influencing student success, like academic performance, personal traits, and family background. In their study, Qureshi and Lokhande [4], explore how different machine learning algorithms, including Random Forest, Decision Trees, and Support Vector Machines (SVM), can be used to make sense of these factors and predict outcomes. They stress the importance of using diverse data to really understand what shapes student performance.

Chandra and Kumar [5] take a slightly different approach, looking into how machine learning can predict student placement in jobs after graduation. They examine the role of academic scores, technical skills, and communication abilities, using data visualisation and preprocessing techniques to understand how these factors contribute to securing a job. Their work demonstrates how academic performance doesn't just influence grades, it also impacts career opportunities, with machine learning helping to predict students' success in landing a job.

Building on this, Ahmed [6] focuses on how machine learning can be used in online learning environments. By examining the interactions students have with learning platforms, he shows how techniques like K-means clustering and Support Vector Machines can help predict student success. This research highlights how these predictions can not only improve outcomes for students but also boost institutional rankings, making a case for the value of machine learning in higher education.

Further, Lagrazon et al. [7] look at licensure exams for Electronics Engineering graduates. Their study shows how combining predictions from several machine learning models, known as ensemble models, can improve the accuracy of predicting exam results. This, in turn, helps educational institutions tweak their curriculum to better prepare students for these exams, ensuring better outcomes in the long run.

Rimpy et al. [8] review a range of EDM techniques used for predicting student performance, highlighting how data mining can identify patterns that help educators understand when students are at risk of poor performance. Early identification allows schools to intervene before students face serious challenges in their academic journey. This research demonstrates the power of data in transforming educational systems and decision-making at all levels.

Asthana et al. [9] add another layer by looking at regression-based models to predict student performance. Their study introduces the concept of 'Learning Coefficients', a measure of a student's potential to learn, which can guide targeted interventions. Using models like Random Forest and Support Vector Regression, they found that linear regression was the most accurate in predicting academic success. This study highlights how regression models can not only predict outcomes but also provide metrics that can be used to improve student performance.

In the world of higher education, Bird [10] discusses the potential of predictive analytics to improve student success. While not diving into specific data, the paper explores the broader picture, showing how machine learning can inform academic strategies and help institutions decide on the best ways to support students. Similarly, Issah et al. [11] provide a systematic review of various machine learning methods that reveal academic and demographic factors are the most important when predicting student performance. However, they also note a gap in research regarding intervention strategies and encourage more work on using predictive models to prevent academic decline.

Oppong [12] provides an overview of machine learning's role in student performance prediction, focusing on the strength of neural networks. His review confirms that supervised learning techniques, especially neural networks, tend to provide the best prediction accuracy, underscoring the importance of choosing the right algorithm for each prediction task.

Meanwhile, Mubarak et al. [13] explore how Graph Convolutional Networks (GCN) can help predict student performance based on their interactions with course materials. By using a semi-supervised approach, they classify students into categories like "high engagement" and "at-risk." This research shows how GCNs can be powerful tools for predicting student success by identifying behavioural patterns and helping educators offer more personalised support.

Finally, it's a groundbreaking approach is presented by Fazil et al. [14], who introduce a deep learning model that considers student behaviour, such as their interaction with virtual learning environments (VLEs). Their system, called ASIST, combines attention mechanisms with convolutional and bidirectional LSTM networks to predict student performance. By processing both behavioural and academic data, ASIST categorises students into different performance groups, allowing educators to make early interventions and help students improve before it's too late.

# 4. Methodology

This section explains the approach taken in this study to predict and assess student performance using machine learning methods. The process covers data collection, preprocessing, feature engineering, model selection, training, evaluation, and result analysis. The methodology was designed to ensure a comprehensive understanding of student performance and to select the most suitable model based on performance metrics.

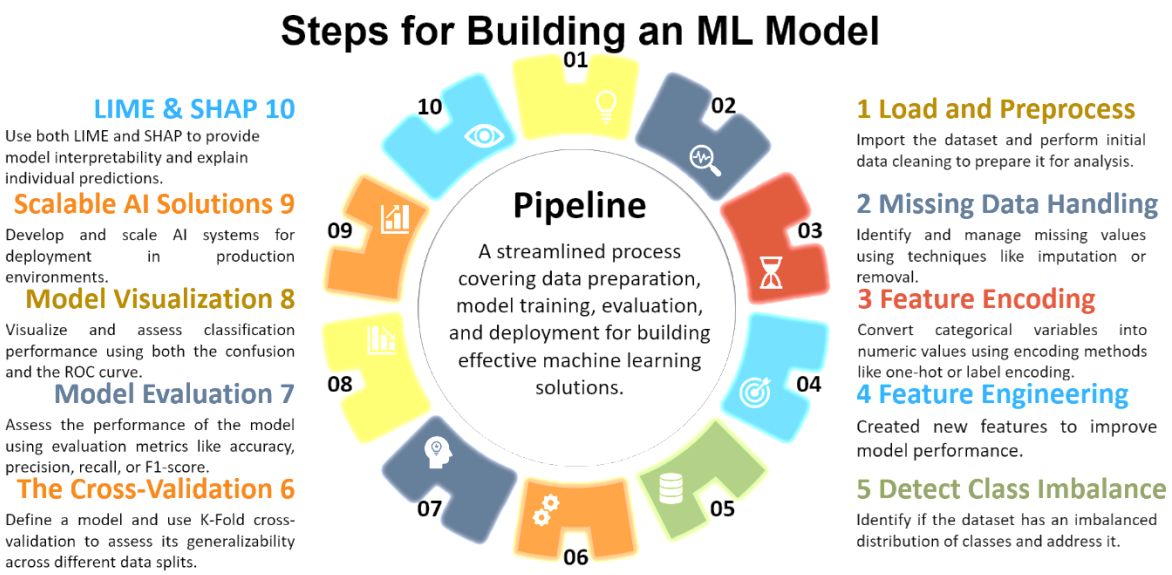


Fig 1: The Pipeline

### 4.1 Data Collection and Dataset Overview

The dataset used for this study is real real-time dataset collected by Dr. Shatha Ghareeb. It consists of various student data points from an educational institution. There are 1500 rows (unique student information), and 30 columns. The data doesn't include any personal information, and the dataset deals with all Artificial Intelligence Ethics, and these columns consist of 2 types of data.

The categorical data i.e., these data points include student demographics (e.g., age, gender), academic records (e.g., previous grades, current grades), attendance records, participation in online forums, assignment submission rates, and other behavioural indicators. This data is essential for evaluating how varied factors impact student performance. This data is very crucial for machine learning models because this provides a direct information about the student's previous and current curricula, including the school and additional information about the student.

| Attribute name | Value | Description |
|---|---|---|
| Gender | Male/Female | Gender of the student |
| Student age (as of 2017/18) | 6, 7, 8, 9,.....etc | Age of the student calculated from 2017/18 academic year |
| Year of admission | Old BBS Student, Old GEMS Student, New Admission 18/19 | The data collected is for two or more academic years; 2017/18 and before academic years + 2018/19 academic year |
| Current year (17/18) | Foundation Stage 1, Foundation Stage 2, Year 1–12/Grade 1–11 | This is the year or grade group assigned to the student by the school |
| Proposed year/grade (18/19) | Foundation Stage 1, Foundation 2, Year 1–13/Grade 1–12 | This is the year or grade group assigned to the student by the school |
| Previous school (17/18) | Many schools in UAE | Previous schools that the student was in before this study |
| Previous curriculum | British/American/MOE/Canadian/Indian/Australian/CBSE/German | The curriculum the student transferred from |
| Current school | GEMS, Belvedere | Name of the school that the data has been collected |
| Current curriculum | Binary value | The curriculum the student transferred to |

Table 1: Categorical features.

Another data type is numerical. First the entrance marks of subjects Mathematics, Science and English are written out of 100 and then for next 2 years (2018-19 and 2019-20), all the 3 terms per year marks are written in percentage. These numerical values are further used in feature engineering this will be discuss in Feature Engineering part methodology section. attendance, quiz scores, etc.). The data is used to predict the students' academic success or failure, and the target variable is typically a categorical label such as "Pass" or "Fail".

| Attribute name | Value | Description |
|---|---|---|
| Math-exam | Mark out of 100 | Exam marks for school entry exam in math |
| Science-exam | Mark out of 100 | Exam marks for school entry exam science |
| English-exam | Mark out of 100 | Exam marks for school entry exam English |
| Math19-1 | Percentage out of 100% | Term 1 student Maths Exam marks during academic year 2018/19 |
| Science19-1 | Percentage out of 100% | Term 1 student science Exam marks during academic year 2018/19 |
| English19-1 | Percentage out of 100% | Term 1 student English Exam marks during academic year 2018/19 |
| Math19-2 | Percentage out of 100% | Term 2 student Maths Exam marks during academic year 2018/19 |
| Science19-2 | Percentage out of 100% | Term 2 student science Exam marks during academic year 2018/19 |
| English19-2 | Percentage out of 100% | Term 2 student English Exam marks during academic year 2018/19 |
| Math19-3 | Percentage out of 100% | Term 3 student Maths Exam marks during academic year 2018/19 |
| Science19-3 | Percentage out of 100% | Term 3 student science Exam marks during academic year 2018/19 |
| English19-3 | Percentage out of 100% | Term 3 student English Exam marks during academic year 2018/19 |
| Math20-1 | Percentage out of 100% | Term 1 student Maths Exam marks during academic year 2019/20 |
| Science20-1 | Percentage out of 100% | Term 1 student science Exam marks during academic year 2019/20 |
| English20-1 | Percentage out of 100% | Term 1 student English Exam marks during academic year 2019/20 |
| Math20-2 | Percentage out of 100% | Term 2 student Maths Exam marks during academic year 2019/20 |
| Science20-2 | Percentage out of 100% | Term 2 student science Exam marks during academic year 2019/20 |
| English20-2 | Percentage out of 100% | Term 2 student English Exam marks during academic year 2019/20 |
| Math20-3 | Percentage out of 100% | Term 3 student Maths Exam marks during academic year 2019/20 |
| Science20-3 | Percentage out of 100% | Term 3 student science Exam marks during academic year 2019/20 |
| English20-3 | Percentage out of 100% | Term 3 student English Exam marks during academic year 2019/20 |

Table 2: Numerical features.

### 4.2 Data Preprocessing

The raw dataset often contains missing values, irrelevant features, or unbalanced classes that can hinder the performance of machine learning models. The preprocessing steps performed are as follows:

- **Cleaning the Data:** The data should be cleaned to remove errors such as misspellings, extra whitespace, or non-alphanumeric characters. This involves writing a Python function to check for and remove any leading or trailing spaces, replacing spaces with underscores, and removing symbols or unnecessary characters. **(Highlighted)** A clean dataset is essential because even small errors can negatively affect the model's performance. The cleaning process varies depending on the dataset, features, and values involved. Therefore, it is important to thoroughly understand the dataset, identify potential issues, and apply the appropriate cleaning function. This ensures the dataset is unique and free from errors.

| Column Name | Missing Values | Percentage |
|---|---|---|
| Science-exam | 3 | 0.1937 |
| English-exam | 1 | 0.0646 |
| English19-1 | 1 | 0.0646 |
| Math19-2 | 1 | 0.0646 |
| Science19-2 | 1 | 0.0646 |
| English19-2 | 2 | 0.1291 |
| Science19-3 | 1 | 0.0646 |
| Math20-1 | 1 | 0.0646 |
| English20-1 | 1 | 0.0646 |
| Math20-2 | 1 | 0.0646 |

Table 3: Missing values in features.

- **Handling Missing Values:** Missing data is a common challenge in real-world datasets. In this study, missing values were handled using imputation techniques. For numerical data, mean imputation was used, replacing missing values with the mean of the respective column. For categorical data, the mode (most frequent value) was used for imputation.

  For further experimentation, missing values were artificially introduced to better understand how models perform under these conditions. Although there were no missing values in categorical features, missing values were still handled for both categorical and numerical data to ensure robustness.

- **Feature Engineering:** Feature engineering modifies existing features or creates new ones to improve model performance. [14], [15] This study created new features by analysing students' marks from the past two years and their entrance exam performance. To establish a meaningful link between the features, the average marks of each student across multiple terms were calculated, including their entrance exam marks.

  This average was then categorized into two groups:

  **Class 0**: Students with an average above 80%

  **Class 1**: Students with an average below 80%

  This categorization is a crucial foundation for the project, as it helps predict which students are at risk and could require additional attention, providing valuable insights for both students and their guardians.

- **Encoding Categorical Features:** Machine learning models require numerical input, so categorical variables such as gender, year of admission, and other categorical labels were encoded using **One-Hot Encoding**. This converts categorical variables into binary vectors, making them compatible with machine learning models without losing information.

| Gender | Age_as_of_Academic_Year_1718 | Current_Year_1718 | Proposed_YearGrade_1819 | Previous_Curriculum_17182 | Current_School | Current_Curriculum | Previous_yearGrade |
|---|---|---|---|---|---|---|---|
| 1 | 4 | 13 | 13 | 0 | 0 | 0 | 0 |
| 1 | 4 | 13 | 13 | 0 | 0 | 0 | 0 |
| 1 | 4 | 13 | 13 | 0 | 0 | 0 | 0 |
| 0 | 4 | 13 | 13 | 0 | 0 | 0 | 0 |
| 1 | 3 | 13 | 13 | 0 | 0 | 0 | 0 |

Table 4: Categorical values converted into numerical values

This step improves model efficiency and performance. It was observed in experiments that categorical data, when directly fed into the model, not only impacted the results but also affected the computational time of models. Although the time difference was in microseconds, it was noticeable and worth addressing.

- **Normalization/Standardization:** Some features in the dataset had different scales, which could negatively affect model performance. To address this, feature scaling was applied using Min-Max Scaling, normalizing the values between 0 and 1. This ensures that all features contribute equally to the learning process, improving the model's overall performance.

- **Feature Selection:** Not all features are equally important for predicting student performance. Feature selection techniques, such as correlation analysis and mutual information, were employed to identify and retain the most relevant features. This reduced the dataset to the most significant features, which improved both accuracy and interpretability.

  *“In the dataset, the "Year_of_Admission" column had three unique values: "School 1 Current Student," "School 2 Current Student," and "New Admission 2019-20." The values for "School 1" and "School 2" were merged into one value: "Current Student" because the "Current School" column already contained this information. Additionally, an imbalance was discovered in the "Current Student" and "New Student" categories. With 1397 rows labeled "Current Student" and only 103 for "New Admission 2019-20," this imbalance could introduce bias into the model. Therefore, 103 rows corresponding to "New Admission 2019-20" were removed using dimensionality reduction techniques, and the "Year_of_Admission" column was subsequently dropped.”*

### 4.3 Model Selection and Training

In this study, several machine learning algorithms were considered for predicting student performance. These include both traditional models and advanced ensemble techniques. The following models were implemented:

1. **LightGBM**: A high-performance gradient boosting framework that is faster and more efficient than traditional models, particularly useful for large datasets.
2. **XGBoost**: A popular gradient boosting technique known for its high accuracy and efficiency, often outperforming other models on structured/tabular data.
3. **Voting Classifier** (Logistic Regression, ANN, SVM, LightGBM): An ensemble model combining predictions from Logistic Regression, Artificial Neural Networks (ANN), SVM, and LightGBM. The final prediction is made based on a majority vote from the individual models.

The models were trained using the pre-processed dataset, and the **training-validation split** was used, where 80% of the data was used for training the model and 20% for validation. The models were then fine-tuned using **hyperparameter optimization techniques** (e.g., Grid Search, Random Search) to identify the best configuration of parameters for each model.

### 4.4 Evaluation Metrics

To assess model performance, several evaluation metrics were employed:

1. **Accuracy**: The ratio of correct predictions to total predictions. While commonly used, accuracy can be misleading with imbalanced datasets.
2. **Precision**: The proportion of true positives (correct predictions of passing students) out of all predicted positive instances.
3. **Recall (Sensitivity)**: The proportion of true positives out of all actual positive instances. This is crucial for identifying students at risk of failure.
4. **F1 Score**: The harmonic mean of precision and recall, balancing the two metrics in situations where both false positives and false negatives are critical.
5. **Confusion Matrix**: A matrix showing actual vs. predicted classifications, providing deeper insights into model errors.

The models were evaluated on the validation set, and the best-performing model was selected for further testing.

### 4.5 Expandable AI (LIME & SHAP)

This study also explores Explainable AI (XAI) techniques, specifically LIME (Local Interpretable Model-agnostic Explanations) and SHAP (SHapley Additive exPlanations), to improve model transparency and interpretability.

- **LIME** explains individual predictions by analysing the specific features associated with an instance [16]. LIME generates an explanation that shows how each feature contributed to the model's prediction, providing valuable insights into model decision-making.
- **SHAP** offers global and local interpretability by attributing the output of a model to individual features, ensuring that every prediction is explainable and transparent [17]. SHAP values allow for a deeper understanding of how each feature influences the final prediction across the entire dataset.

These techniques make it easier for both developers and end-users to trust the model, ensuring that decisions are based on clear, understandable reasoning.

**Summary**
By utilizing LIME and SHAP, alongside traditional and ensemble models such as LightGBM and XGBoost, this study enhances model transparency, interpretability, and performance. These tools help ensure that predictions are not only accurate but also explainable, facilitating trust in machine learning-based decisions.

### 4.6 Introduce a Web-Application

Integrating a web application into this study adds a practical aspect, allowing users students, parents, and educators to input data such as demographic details, academic performance, and exam scores. The app then provides personalised predictions, helping to answer the question: *"Does the student's chosen grade align with their interests?"*

Designed to be intuitive and user-friendly, the app guides users through the process of entering data. Once submitted, it communicates with the machine learning models running in the backend, returning predictions about the student's academic future, along with tailored improvement suggestions.

Benefits for Users:

- Students can use the app for self-assessment, identifying areas where they need to focus more.
- Parents gain insight into their child's academic path, allowing them to offer timely support.
- Schools can identify at-risk students early, taking proactive measures to improve their learning experience.

By incorporating the web app, the study transitions from theoretical models to real-world applications, making the insights more accessible and actionable for all involved. The app not only enhances learning outcomes but also empowers users to make informed decisions.

# 5. Results

The results obtained from the machine learning models implemented in this study. Several performance metrics were used to evaluate the models, including accuracy, precision, recall, F1 score, and the confusion matrix.

### 5.1 Data Pre-Processing Result

| Class Distribution in Training Data | |
|---|---|
| **Class** | **Count** |
| 0 | 594 |
| 1 | 524 |
| **Total** | **1118** |

Table 5: Class distribution for Training data.

The training data was balanced after the application of Principal Component Analysis (PCA) for dimensionality reduction. This ensured that the dataset had an even distribution across classes, addressing any previous imbalance.

| Class Distribution in Test Data | |
|---|---|
| **Class** | **Count** |
| 0 | 148 |
| 1 | 131 |
| **Total** | **279** |

Table 6: Class distribution for Testing data.

Similarly, the testing data was also balanced, following the same preprocessing steps. The total dataset of 1,397 instances was split into a 4:1 ratio for training and testing. This division further confirmed that class imbalance was effectively removed. Initially, the dataset had an imbalance, especially concerning new students, with only 103 records for them. This imbalance could have led to bias in the model. However, after eliminating the rows with such records and removing the "Year_Of_Admission" feature, this bias was rectified. As emphasised by experts, the larger the dataset, the better the performance of a model, but it is crucial that the data is free of any biases. Any bias in the data could negatively impact the model's performance and lead to unreliable predictions on real-time data. Hence, ensuring that the dataset is balanced and unbiased is vital for accurate model predictions.

### 5.2 Performance Metrics

The choice of performance metrics depends on the model type, whether it is a classification or regression task. In this study, the models were evaluated using the following key metrics: Accuracy, Precision, Recall, F1 Score, and ROC AUC. These metrics provide insights into different aspects of model performance. The table below summarises the results for each of the models.

| Model | Accuracy | F1-Score | Precision | Recall |
|---|---|---|---|---|
| **LightGBM** | 0.899 | 0.89 | 0.912 | 0.869 |
| **XGBoost** | 0.893 | 0.883 | 0.906 | 0.861 |
| **Voting Classifier** | 0.929 | 0.923 | 0.941 | 0.907 |

Table 7: Performance metrics for each model.

- The Voting Classifier performed the best among the models, achieving the highest accuracy of 0.929 and the best F1 Score of 0.923. It also demonstrated the highest precision (0.941) and recall (0.907), indicating that it was the most balanced in correctly identifying both positive and negative cases.
- The LightGBM model achieved a good performance with an accuracy of 0.899, a precision of 0.912, and an F1 Score of 0.89. However, its recall of 0.869 was slightly

lower than that of the Voting Classifier, meaning it missed a few positive cases compared to the other models.

- XGBoost, while showing strong performance with an accuracy of 0.893, a precision of 0.906, and an F1 Score of 0.883, had a slightly lower recall (0.861) than both LightGBM and the Voting Classifier.

### 5.3 ROC AUC Analysis

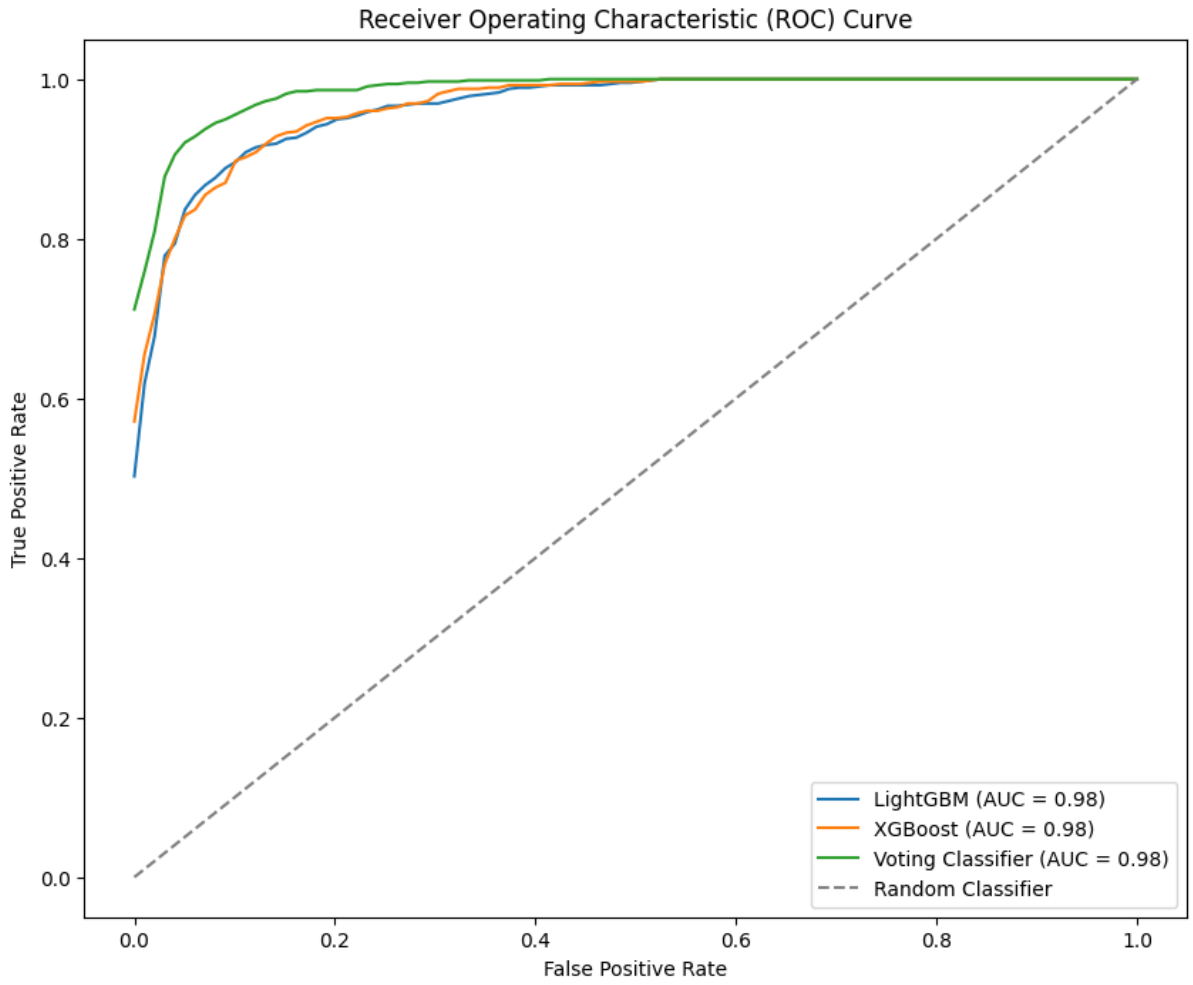


Fig 2: The ROC curve for each model.

| Model | LightGBM | XGBoost | Voting Classifier |
|---|---|---|---|
| ROC AUC | 0.965 | 0.966 | 0.985 |

Table 8: Tabular representation of ROC curve for each model.

The Voting Classifier had the highest ROC AUC of 0.985, indicating it had the best ability to correctly classify both positive and negative cases across all thresholds. Both LightGBM and XGBoost performed similarly, with ROC AUC values of 0.965 and 0.966, respectively, showing they were also strong models, but slightly less effective in distinguishing between classes compared to the Voting Classifier.

### 5.4 Confusion Matrix Analysis

The confusion matrix provides an overview of how well each model is performing in terms of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN). These metrics are important for evaluating the model's classification performance, especially when dealing with imbalanced datasets.

| Model | FP | FN | TP | TN |
|---|---|---|---|---|
| **LightGBM** | 6 | 15 | 116 | 142 |
| **XGBoost** | 6 | 19 | 112 | 142 |
| **Voting Classifier** | 3 | 11 | 120 | 145 |

Table 9: The confusion matrix representation for each model.

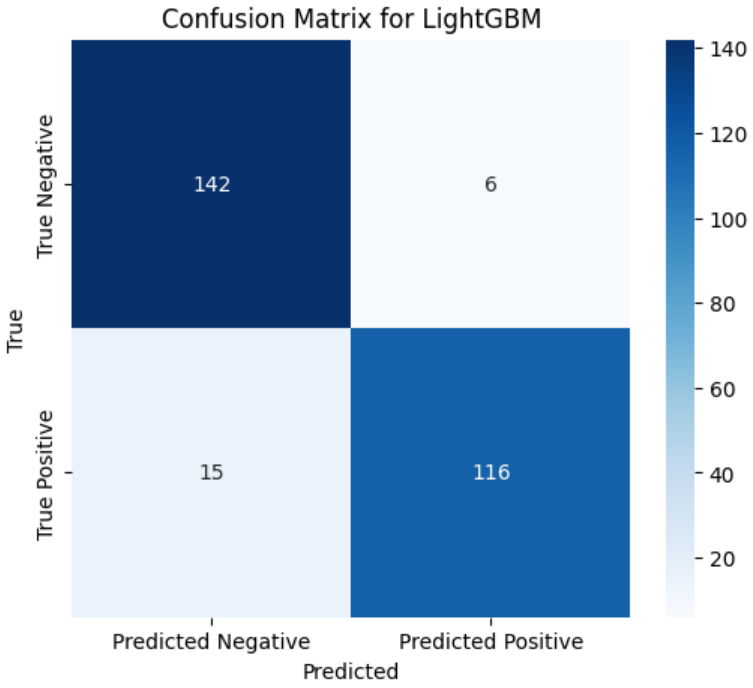


Fig 3: Confusion Matrix for LightGBM

The LightGBM model had 6 false positives, which means 6 instances were incorrectly classified as positive when they were actually negative. It also had 15 false negatives, where the model missed 15 positive cases, classifying them as negative. However, LightGBM performed well in terms of true positives (116) and true negatives (142), correctly identifying most of the positive and negative instances. The relatively low number of false positives and false negatives indicates that the model was fairly accurate, although it could still be improved by reducing the number of false negatives, which would result in fewer missed positive cases.

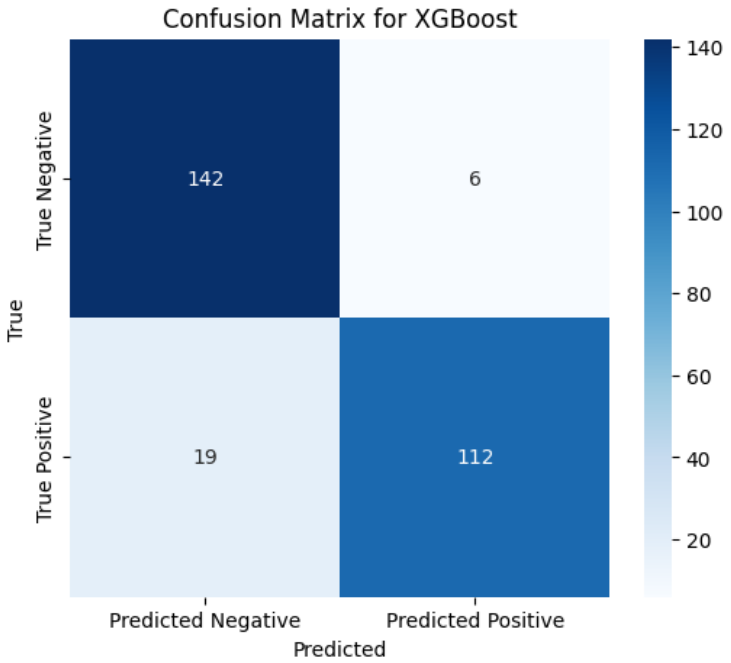


Fig 4: Confusion Matrix for XGBoost

For the XGBoost model, there were 6 false positives, the same as LightGBM, but the number of false negatives increased to 19, indicating that this model missed more positive instances compared to LightGBM. The true positive count was 112, and the true negatives were 142, which is like LightGBM. While the model performed fairly well overall, the increased false negatives suggest that XGBoost may benefit from adjustments to its classification threshold or further tuning to reduce missed positive cases.

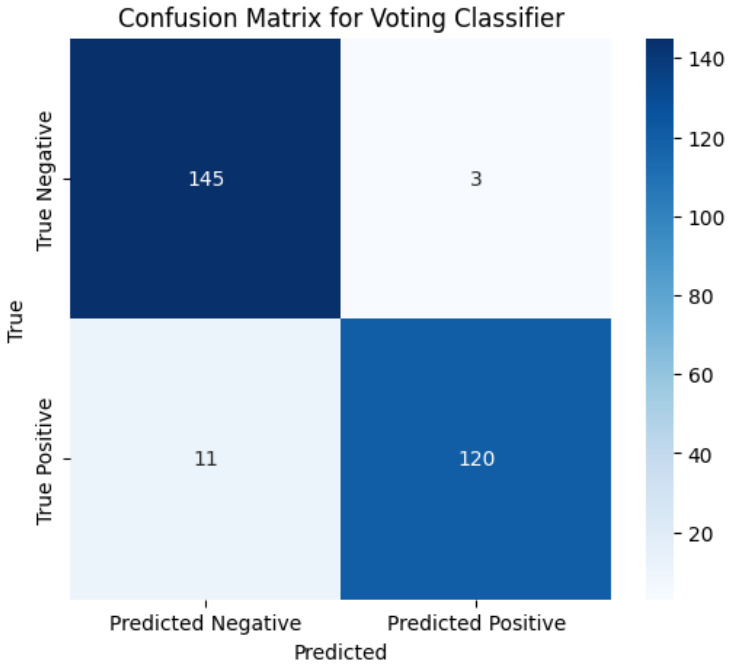


Fig 5: Confusion Matrix for Voting Classifier.

The Voting Classifier showed the best performance among the models in terms of the confusion matrix. It had the lowest number of false positives (3), meaning it made fewer

incorrect positive classifications. It also had 11 false negatives, which is the lowest among the three models, indicating that it was the best at identifying positive cases. With 120 true positives and 145 true negatives, the Voting Classifier demonstrated a strong ability to correctly identify both positive and negative instances. Overall, this model exhibited the most balanced performance, with fewer misclassifications compared to the other models.

.

## 5.5 Expandable AI

### 5.5.1 Local Interpretable Model-agnostic Explanations (LIME) Result

As this study also focuses on the interpretability aspect of AI models using LIME. Since each model processes data differently, the impact of specific features can vary from one model to another. For this study, instance number 113 was selected (row 114 in the dataset) to examine how different features influence predictions across multiple models. The table below presents the LIME results, highlighting which features had the most positive or negative effect on the prediction for that instance.

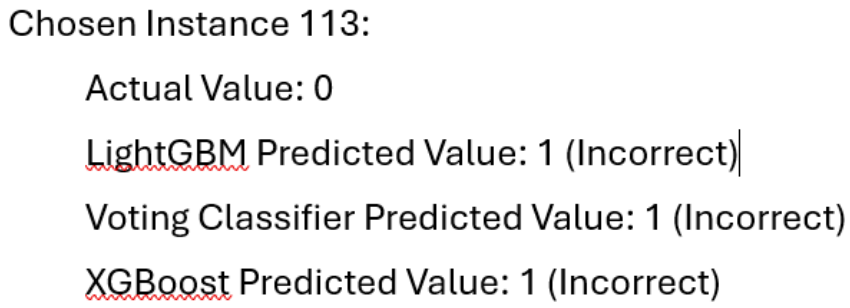


Fig 6: LIME setup for instance 113.

The instance 113 has 1 actual value but all the models predict it wrongly as 0. By using LIME, the study will enhance about the reason to for the model's prediction.

1. **LightGBM Model**

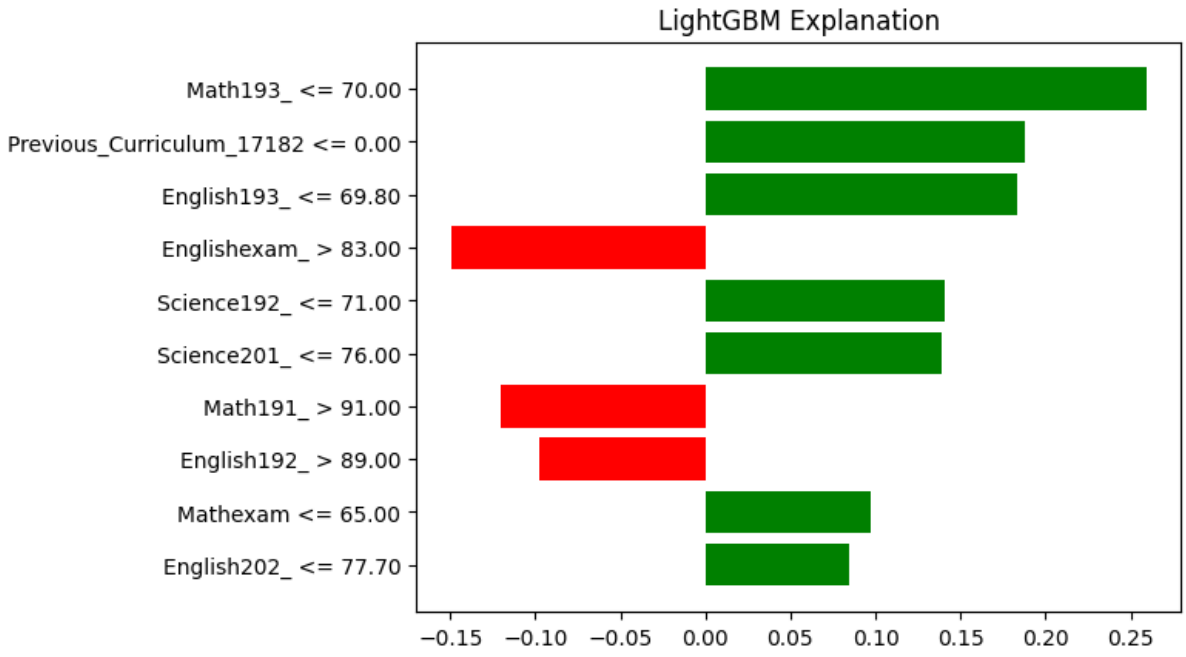


Fig 7: LIME explanation for LightGBM Model.

The LightGBM model identifies Math193_ <= 70.00 as the most influential feature with a strong positive impact (0.2993) on the prediction. It indicates that scoring 70 or less in Math193 increases the likelihood of the predicted outcome. Other positively influential features include low scores in English193, Science192, and Science201, and Mathexam <= 65.00.

Negative influences come from high scores like Englishexam_ > 83.00, Math191_ > 91.00, and English192_ > 89.00. These seem to reduce the likelihood of the predicted outcome. The mix of positive and negative contributions shows the model considers both strengths and weaknesses in academic scores when making a prediction.

2. **Voting Classifier**

As per the reference by [18], a Voting Classifier combines multiple machine learning models to improve prediction accuracy by considering their collective votes. This helps make the model more robust and reduces the risk of overfitting.

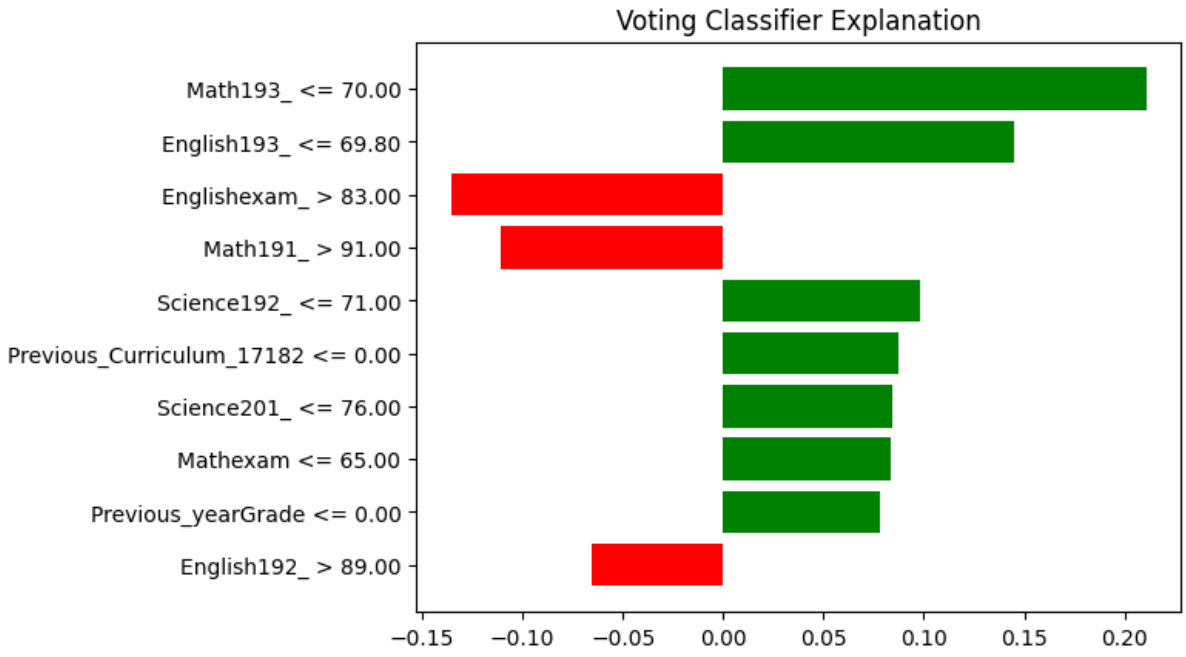


Fig 8: LIME explanation for Voting Classifier Model.

This ensemble model also finds Math193_ <= 70.00 as the top positive contributor (0.2132), followed by English193_ <= 69.80 and low Science scores. These lower academic scores in specific subjects are pushing the prediction positively.

Negative impacts are seen with high Englishexam (> 83.00) and Math191 > 91.00, as well as English192 > 89.00, which slightly lowers the probability of the outcome. The Voting Classifier presents a balanced view of the student's performance, leaning more on weaker scores as decision drivers.

### 3. XGBoost

By referring to the work of Li and Zhou (2023) [19], which helps to gain a deeper understanding of the XGBoost algorithm.

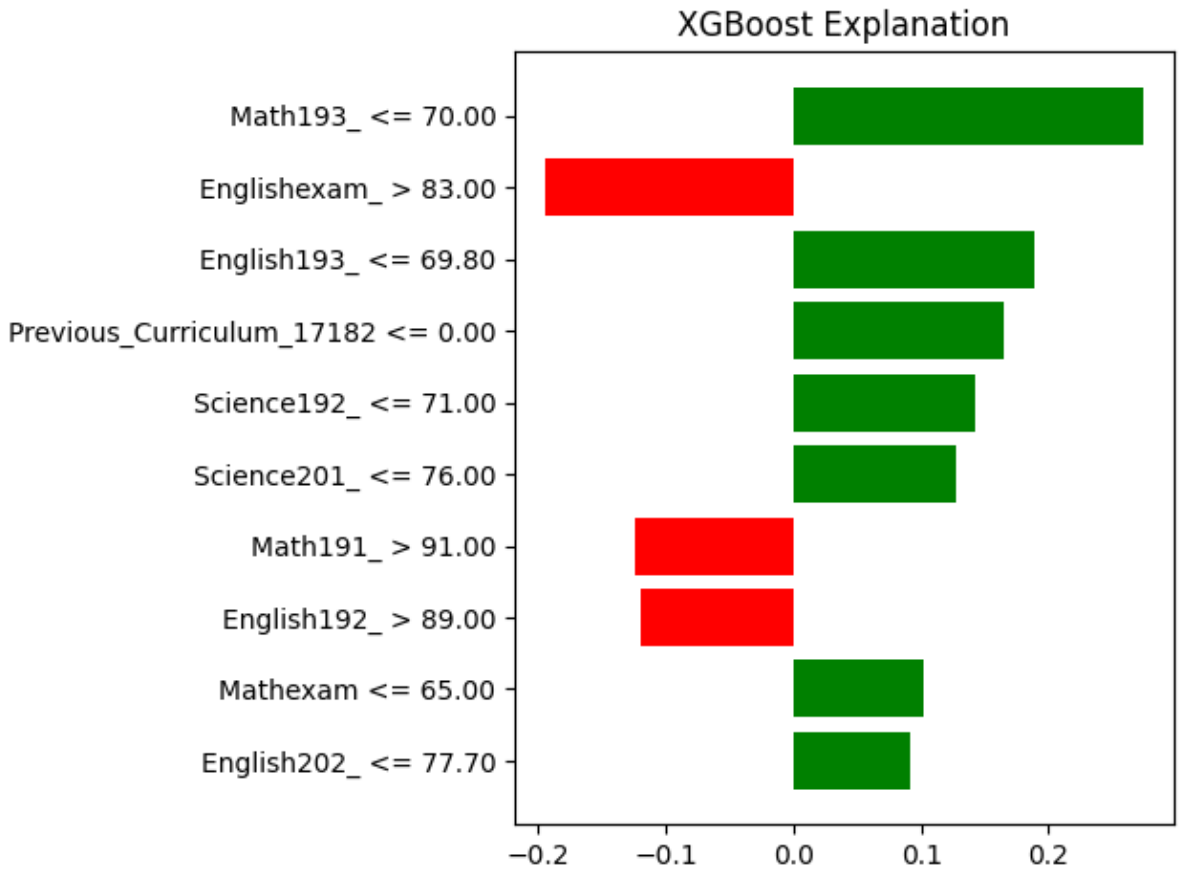


Fig 9: LIME explanation for XGBoost Model.

For the XGBoost model, again, Math193_ <= 70.00 is the most significant positive feature (0.2602), like the other models. Low scores in English193, Science201, Science192, and Mathexam also positively influence the result.

However, higher scores in Englishexam_ > 83.00, Math191 > 91.00, and English192 > 89.00 negatively affect the outcome, suggesting that strong academic performance in these areas reduces the model's confidence in the predicted class.

This model appears slightly more sensitive to exam performance than the Voting Classifier.

### 5.5.2 SHapley Additive exPlanations (SHAP) Result

SHAP (SHapley Additive exPlanations) helps to understand how much each feature contributes to a model's predictions. A higher SHAP value means the feature has more influence. Below is a breakdown of the SHAP importance values for the LightGBM and XGBoost models.

| SHAP Importance for Model | | |
|---|---|---|
| Feature | LightGBM | XGBoost |
| Gender | 0.02162 | 0.03251 |
| Age_as_of_Academic_Year_1718 | 0.05042 | 0.03475 |
| Current_Year_1718 | 0.01965 | 0.03665 |
| Proposed_YearGrade_1819 | 0.01065 | 0.03717 |
| Previous_Curriculum_17182 | 0.81216 | 0.85990 |
| Current_School | 0.00000 | 0.00000 |
| Current_Curriculum | 0.00000 | 0.00000 |
| Previous_yearGrade | 0.00000 | 0.00000 |
| Mathexam | 0.48087 | 0.65687 |
| Scienceexam_ | 0.39440 | 0.41421 |
| Englishexam_ | 0.70947 | 0.91023 |
| Math191_ | 0.68265 | 0.77959 |
| Science191_ | 0.49839 | 0.55214 |
| English191_ | 0.39336 | 0.48451 |
| Math192_ | 0.45452 | 0.59257 |
| Science192_ | 0.57827 | 0.69380 |
| English192_ | 0.61541 | 0.69326 |
| Math193_ | 1.15292 | 1.25502 |
| Science193_ | 0.39981 | 0.50626 |
| English193_ | 0.68885 | 0.83583 |
| Math201_ | 0.50894 | 0.63404 |
| Science201_ | 0.84779 | 0.96285 |
| English201_ | 0.38489 | 0.55300 |
| Math202_ | 0.38964 | 0.52740 |
| Science202_ | 0.25580 | 0.37572 |
| English202_ | 0.39570 | 0.46495 |
| Math203_ | 0.69594 | 0.81393 |
| Science203_ | 0.45082 | 0.51704 |
| English203_ | 0.43362 | 0.57695 |

Table 9: SHAP Importance for Models.

### 1. LightGBM Model

In the LightGBM model, Math193_ is the most influential feature (1.15292), showing it plays a key role in driving the prediction. This is followed closely by Previous_Curriculum_17182 (0.81216) and Science201_ (0.84779), which are also major contributors.

Other strong features include:

- Englishexam_: 0.70947
- Math191_: 0.68265
- Math203_: 0.69594
- English193_: 0.68885

These values suggest that performance in recent academic years, particularly Maths and English, strongly shapes the model's decision.

Lesser influences include Gender and Age_as_of_Academic_Year_1718, while Current_School, Current_Curriculum, and Previous_yearGrade have no measurable impact (0.00000).

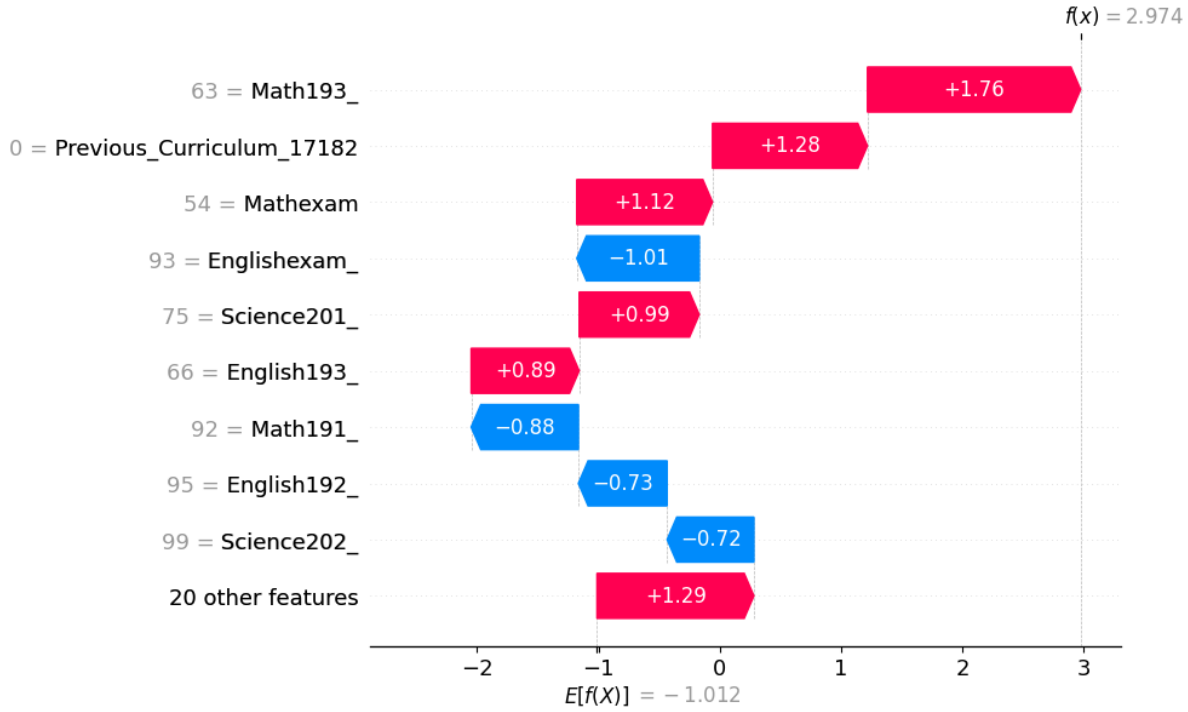


Fig 10: SHAP explanation for LightGBM Model.

### 2. XGBoost Model

For XGBoost, Math193_ again stands out as the top feature (1.25502), indicating it has the highest impact on predictions. This is followed closely by:

- Previous_Curriculum_17182: 0.85990
- Science201_: 0.96285

- Englishexam_: 0.91023
- English193_: 0.83583

These results show that both curriculum history and subject-specific scores in English, Science, and Maths are vital for the model's output.

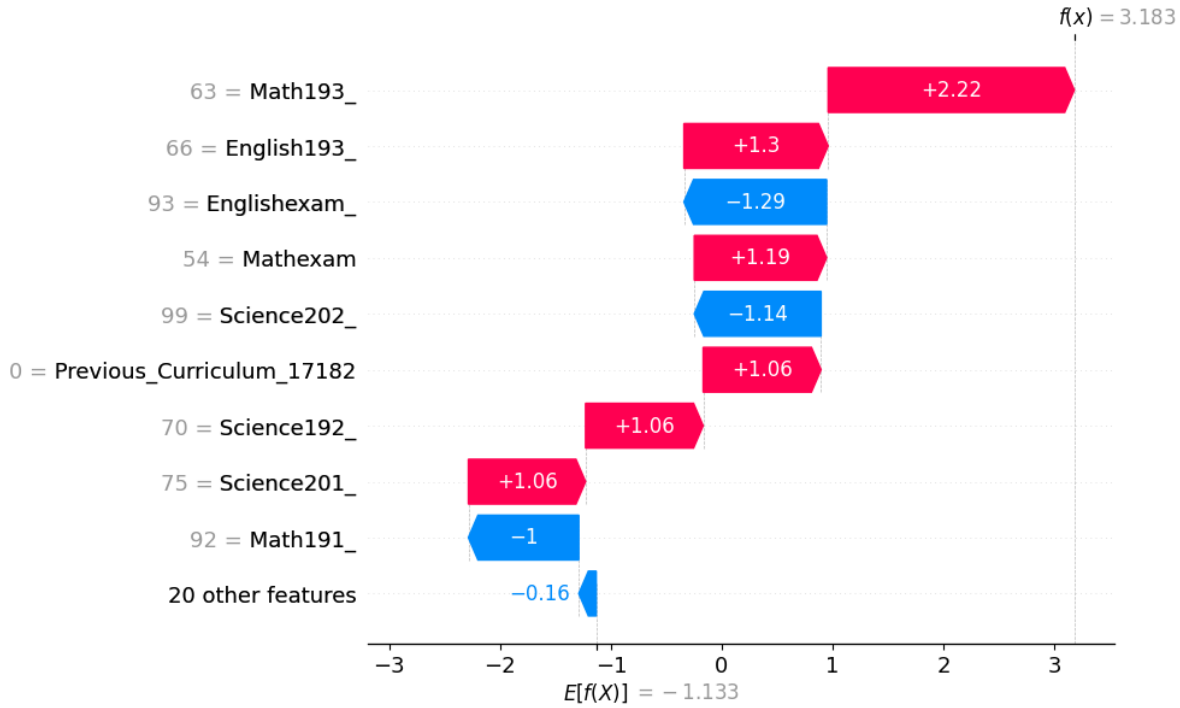


Fig 11: SHAP explanation for XGBoost Model.

Similar to LightGBM, demographic features like Gender, Age, and Current_Year have minimal influence. Again, features like Current_School, Current_Curriculum, and Previous_yearGrade have no effect on the prediction (all are 0.00000).

## 5.6 The Web Application

The web application developed as part of this study uses the Voting Classifier, which was chosen due to its strong performance in predicting student outcomes. After testing multiple machine learning models, Voting Classifier proved to be the most reliable, providing high accuracy and consistency in predicting academic performance based on a combination of numerical and categorical data.

To further validate the model's effectiveness, 500 random new data points were generated and tested using the Voting Classifier model. Of these, 321 instances predicted that the chosen grade was appropriate for the student, while 179 instances indicated potential issues with the grade selection. The significant difference between these two groups highlights how often the model's predicted grade aligns with the student's other academic data and subject scores.

| 500 Random New dataset | |
|---|---|
| Prediction by SVM model | Number of Instance |
| In favour | 321 |
| Not in favour | 179 |
| **Total** | **500** |

Fig 12: 500 Random New Dataset for WebApp.

The decision to use the SVM model in the web app was reinforced by the findings from, which showed that the numerical features, such as student scores in Maths, Science, and English, had a larger impact on the model's predictions than the categorical features. This insight confirmed that the SVM model is effective at handling key academic data, making it the most suitable choice for providing accurate predictions.

The web app allows students, parents, and educators to input student data and receive immediate feedback on whether the proposed grade is appropriate. The app not only provides predictions but also offers transparency by explaining which factors contributed to the result. This helps users understand the basis of the predictions, building trust in the model's accuracy.

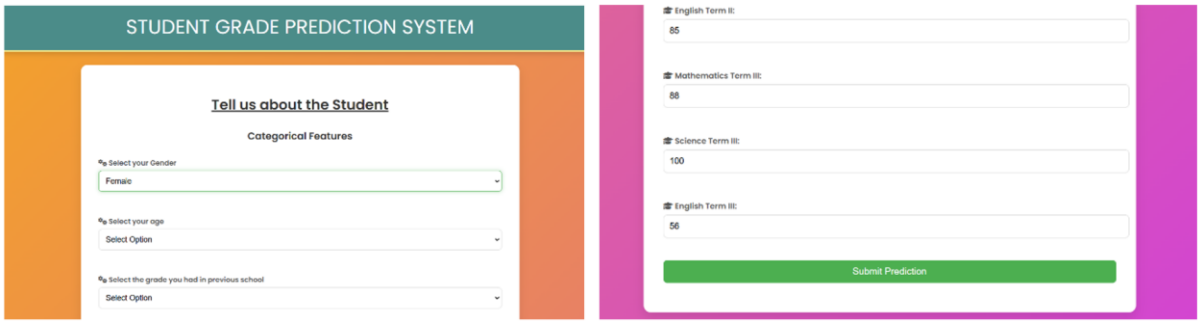


Fig 13: WebApp User input pages

The User can input all the required details about the student in this page, and submit the form for model evaluation, at backend the model will predict the outcome.

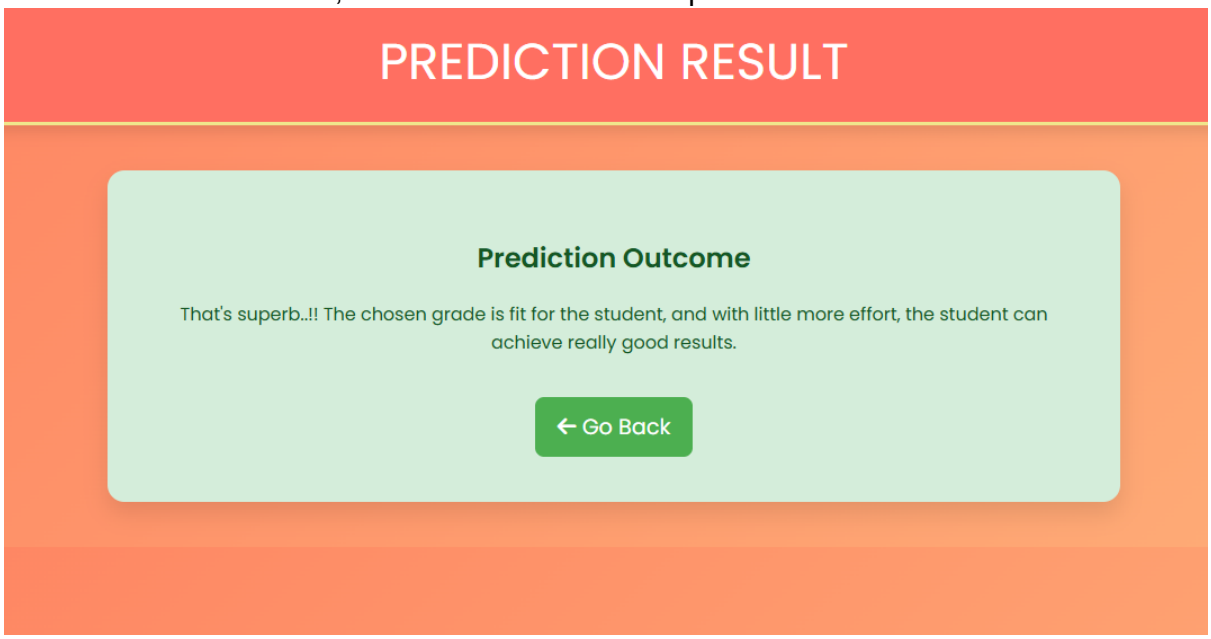


Fig 14: WebApp Result shown after model prediction

The model outcome is dependent upon the scores of students and categorical data, if the chose grade is matching with the Model output them it will says the positive part else the model will suggest like which subject is needed to be improve or user have option to choose the different grade for the student.

By integrating the voting classifier, the web app offers an intuitive and informative tool for all users. It gives students personalised insights into their academic progress, helps parents make informed decisions, and assists schools in identifying students who may need additional support.

# 6. Conclusion and Future Work

This study explored the use of machine learning models to predict student academic outcomes, focusing on accuracy, fairness, and interpretability. By applying proper data pre-processing techniques, including Principal Component Analysis (PCA), the dataset was balanced to ensure more reliable and unbiased results. The original class imbalance, especially concerning new students, was addressed by removing biased records and irrelevant features, improving the quality of the input data.

Among the models tested, the Voting Classifier delivered the best overall performance. It achieved the highest accuracy (0.929), precision (0.941), and F1 Score (0.923), and showed strong results across all evaluation metrics. LightGBM and XGBoost also performed well, though they exhibited slightly higher false negatives, indicating a few more missed positive cases. Nonetheless, all models proved suitable for predicting academic risks with reasonable confidence.

To enhance the transparency of these models, LIME and SHAP were used to interpret the results. Both tools identified Math193, Science201, and English193 as key features influencing predictions. Lower scores in these subjects increased the likelihood of a negative outcome,

while stronger academic results reduced that risk. These insights can help educators understand why specific predictions are made and support more informed interventions.

Looking ahead, there are several opportunities for future work. Expanding the dataset would allow for greater model generalisation and more robust training. Including additional features such as attendance, extra-curricular involvement, and socio-economic background could further improve accuracy and provide a deeper view of student performance. Testing other ensemble methods or neural networks may also yield better results, particularly with larger and more complex datasets.

Finally, more advanced interpretability techniques, like counterfactual explanations or causal models, could offer greater clarity and trust in predictions. These tools would make the models more useful in real-world educational environments, supporting fairer and more timely student support strategies.

In summary, machine learning can play a valuable role in education, provided the models are accurate, interpretable, and free from bias.

# 7. List of Tables and Figures

## 7.1 List of Figures


Fig 1: The Pipeline.

Fig 2: The ROC curve for each model.

Fig 3: Confusion Matrix for LightGBM.

Fig 4: Confusion Matrix for XGBoost.

Fig 5: Confusion Matrix for Voting Classifier.

Fig 6: LIME setup for instance 219.

Fig 7: LIME explanation for LightGBM Model.

Fig 8: LIME explanation for Voting Classifier Model.

Fig 9: LIME explanation for XGBoost Model.

Fig 10: SHAP explanation for LightGBM Model.

Fig 11: SHAP explanation for XGBoost Model.

Fig 12: 500 Random New Dataset for WebApp.

Fig 13: WebApp User input pages.

Fig 14: WebApp Result shown after model prediction.


## 7.2 List of Table


Table 1: Categorical features.

Table 2: Numerical features.

Table 3: Missing values in features.

Table 4: Categorical values converted into numerical values.

Table 5: Class distribution for Training data.

Table 6: Class distribution for Testing data.

Table 7: Performance metrics for each model.

Table 8: Tabular representation of ROC curve for each model.

Table 9: The confusion matrix representation for each model.

Table 10: SHAP Importance for Model.

# 8. References


1. Ghareeb, S., Hussain, A., Khan, W. and Al-Jumeily, D. (2021). Dataset of student level prediction in UAE. *Data in Brief*, [online] 35, p.106908. doi:https://doi.org/10.1016/j.dib.2021.106908.

2. Ghareeb, S., Hussain, A.J., Al-Jumeily, D., Khan, W., Al-Jumeily, R., Baker, T., Al Shammaa, A. and Khalaf, M., 2022. Evaluating student levelling based on machine learning model's performance. *Discover Internet of Things*, 2(1), p.3. doi:https://doi.org/10.1007/s43926-022-00023-0.

3. Pande, S.M. (2023). Machine Learning Models for Student Performance Prediction. *Machine Learning Models for Student Performance Prediction*, [online] pp.27–32. doi: https://doi.org/10.1109/icidca56705.2023.10099503

4. Qureshi, R. and Lokhande, P.S. (2024). A Comprehensive Review of Machine Learning techniques used for Designing An Academic Result Predictor And Identifying The Multi-Dimensional Factors Affecting Student's Academic Results. [online] pp.1–6. doi:https://doi.org/10.1109/idicaiei61867.2024.10842901.

5. Chandra S. K and K Santhosh Kumar (2022). Data Preprocessing and Visualizations Using Machine Learning for Student Placement Prediction. 2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS). doi:https://doi.org/10.1109/ictacs56270.2022.9988247

6. Ahmed, E. (2024). Student Performance Prediction Using Machine Learning Algorithms. Applied Computational Intelligence and Soft Computing, [online] 2024, p.e4067721. doi:https://doi.org/10.1155/2024/4067721.

7. Lagrazon, G.G., Edytha, J., Rossana, M. and Maaliw, R.R. (2023). Ensemble-Based Prediction Model for Enhanced Electronics Engineering Licensure Examination Results Using Student Performance Analysis. [online] doi:https://doi.org/10.1109/iceeie59078.2023.10334657.

8. Rimpy, Dhankhar, A. and Solanki, K. (2022). Educational Data Mining tools and Techniques used for Prediction of Student's Performance: A Study. 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), [online] pp.1–5. doi:https://doi.org/10.1109/icrito56286.2022.9965023.

9. Asthana, P., Mishra, S., Gupta, N., Derawi, M. and Kumar, A. (2023). Prediction of Student's Performance With Learning Coefficients Using Regression Based Machine

Learning Models. *IEEE Access*, 11, pp.72732–72742. doi:https://doi.org/10.1109/access.2023.3294700.

10. Bird, K. (2023). Predictive Analytics in Higher Education: The Promises and Challenges of Using Machine Learning to Improve Student Success. *AIR Professional File*, [online] (Fall 2023). doi:https://doi.org/10.34315/apf1612023.

11. Issah, I., Appiah, O., Appiahene, P. and Inusah, F. (2023). A systematic review of the literature on machine learning application of determining the attributes influencing academic performance. *Decision Analytics Journal*, 7, p.100204. doi:https://doi.org/10.1016/j.dajour.2023.100204.

12. Stephen Opoku Oppong (2023). Predicting Students' Performance Using Machine Learning Algorithms: A Review. *Asian Journal of Research in Computer Science*, 16(3), pp.128–148. doi:https://doi.org/10.9734/ajrcos/2023/v16i3351.

13. Mubarak, A.T., Cao, H., Hezam, I.M. and Hao, F. (2022). Modeling students' performance using graph convolutional networks. 8(3), pp.2183–2201. doi:https://doi.org/10.1007/s40747-022-00647-3

14. Mohd Fazil, Angélica Rísquez and Halpin, C. (2024). A Novel Deep Learning Model for Student Performance Prediction Using Engagement Data. *Journal of learning analytics*, pp.1–19. doi:https://doi.org/10.18608/jla.2024.7985.

15. S, A., V, D., S, M.S. and Srikanth, R. (2023). Systematic Review on Real-Time Students Behavior Monitoring using Machine Learning. *2023 International Conference on Inventive Computation Technologies (ICICT)*, pp.233–237. doi:https://doi.org/10.1109/icict57646.2023.10134519.

16. S., P.G., Dinesh, G., Gupta, D. and Nair, A.R. (2025). Predicting Student Success in Online Learning Using Machine Learning and Explainable AI. 2025 3rd International Conference on Advancement in Computation & Computer Technologies (InCACCT), [online] pp.566–572. doi:https://doi.org/10.1109/incacct65424.2025.11011419

17. Ahmed, S., M. Shamim Kaiser, Mohammad Shahadat Hossain and Andersson, K. (2024). A Comparative Analysis of LIME and SHAP Interpreters with Explainable ML-Based Diabetes Predictions. IEEE Access, pp.1–1. doi:https://doi.org/10.1109/access.2024.3422319

18. Mohammad T; Kahakashan A; Md.Hamid H; Sadia N; Safa A (2024). Predictive Modelling of Anxiety Levels in Bangladeshi University Students: A Voting-Based Approach with LIME and SHAP Explanations. (2024). *Ieee.org*. [online] doi:https://doi.org/10.1109/iCACCESS61735.2024.10499576.

19. M Li, G. and Zhou, H. (2023). Modeling and Estimation Methods for Student Achievement Recognition Based on XGBoost Algorithm. *XGBoost Algorithm*, [online] pp.1–6. doi:https://doi.org/10.1109/easct59475.2023.10392502

20. Luza, J.C.J. and Rodriguez, C. (2024). Predictive Attributes in Machine Learning for University Academic Performance: A Feature Engineering Approach. *2024 IEEE 16th International Conference on Computational Intelligence and Communication Networks (CICN)*, [online] pp.443–456. doi:https://doi.org/10.1109/cicn63059.2024.10847424.