Department of Computer Science & Engineering,
School of Computer Science and Engineering,
Manipal University Jaipur,
February 2025

A Report

On

Empirical Evaluation of Classical Machine Learning Models for Predicting Student Performance in Online Education

carried out as part of the course Minor Project- AI 3270

Submitted by

Aryan Jhamnani
229309143
Arham D Jain
229310146
VI-AIML

in partial fulfilment for the award of the degree

of
BACHELOR OF TECHNOLOGY

In

Computer Science & Engineering

Under the Guidance of :

Guide Name: ***Dr. Surendra Solanki***

Acknowledgement

We would like to express my heartfelt gratitude to the Dean, School of Computer Science and Engineering, Manipal University Jaipur, for providing the academic environment and necessary infrastructure for the successful completion of this project. I am sincerely thankful to the Associate Dean, School of Computer Science and Engineering, for his/her valuable guidance and administrative support throughout the project duration. I deeply appreciate **Dr. Deepak Panwar**, Head of the Department of Artificial Intelligence and Machine Learning, for enabling a conducive research atmosphere and constant encouragement. My special thanks to my project supervisor, **Mr. Surendra Solanki**, for his expert supervision, insightful feedback, and unwavering support during the research and documentation process. Lastly, I am grateful to all the faculty members and staff of the Department of AIML for their cooperation throughout this project.

229309143 **Aryan Jhamnani**
229310146 **Arham D Jain**

Guide Signature (with date): _____

MANIPAL UNIVERSITY
JAIPUR
(University under Section 2(f) of the UGC Act)

Department of Computer Science and Engineering
School of Computer Science and Engineering

Date: _____

CERTIFICATE

This is to certify that the project entitled "Empirical Evaluation of Classical Machine Learning Models for Predicting Student Performance in Online Education" is a Bonafide work carried out as part of the course AI3270, under my guidance from Jan 2025 to May 2025 by Arham Jain(229310146), student of B. Tech (hons.) Computer Science and Engineering (AIML), 6th Semester at the Department of Artificial Intelligence and Machine Learning, Manipal University Jaipur, during the academic semester 6th in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering (AIML), at MUJ, Jaipur.

Mr. Surendra Solanki
Project Guide, Dept. of AIML
Manipal University Jaipur

Dr. Deepak Panwar
HOD, Dept. of AIML
Manipal University Jaipur

MANIPAL UNIVERSITY
JAIPUR
(University under Section 2(f) of the UGC Act)

Department of Computer Science and Engineering
School of Computer Science and Engineering

Date: _____

CERTIFICATE

This is to certify that the project entitled "Empirical Evaluation of Classical Machine Learning Models for Predicting Student Performance in Online Education" is a Bonafide work carried out as part of the course AI3270, under my guidance from Jan 2025 to May 2025 by Aryan Jhamnani(229309143), student of B. Tech (hons.) Computer Science and Engineering (AIML), 6th Semester at the Department of Artificial Intelligence and Machine Learning, Manipal University Jaipur, during the academic semester 6th in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering (AIML), at MUJ, Jaipur.

Mr. Surendra Solanki
Project Guide, Dept. of AIML
Manipal University Jaipur

Dr. Deepak Panwar
HOD, Dept. of AIML
Manipal University Jaipur

Abstract

Student engagement is a critical factor in the success of online learning, influencing academic performance, retention, and overall learner outcomes. Traditional engagement tracking methods rely on subjective and delayed assessments, making early interventions difficult. This study investigates the use of machine learning to predict student engagement using behavioral and academic data from online platforms.

We initially evaluate three models—Multi-Layer Perceptron (MLP), Long Short-Term Memory (LSTM), and XGBoost—on the Open University Learning Analytics Dataset (OULAD). MLP demonstrated poor generalization, achieving only 56.93% test accuracy. LSTM performed better, leveraging sequential patterns to achieve 74.10% accuracy. However, XGBoost offered a strong balance between performance (71.61% accuracy), interpretability, and computational efficiency.

Due to architectural differences between deep learning and tree-based models, we selected XGBoost as a robust baseline for further comparison with classical machine learning models including Logistic Regression (L1, L2), Support Vector Machines (SVM), Random Forest, and Gradient Boosting. To simplify the classification task and improve model reliability, we transitioned from a multi-class setup to a binary Pass/Fail prediction using a 40% score threshold.

Results showed that Logistic Regression and SVM models achieved comparable performance to ensemble models, with accuracies exceeding 96% across various train-validation-test splits (80/10/10, 70/15/15, 60/20/20). Feature importance analysis highlighted variables like sum of clicks, studied credits, and past assessment scores as strong predictors of student engagement and success..

Keywords

Student Engagement, Machine Learning, XGBoost, LSTM, Online Learning Analytics, Logistic Regression, Education Data Mining

Table of Contents

1. Introduction

1.1 Background

With the rapid expansion of online education, student engagement has become a crucial factor in determining academic success. Engagement in online learning refers to the level of interaction, participation, and attentiveness demonstrated by students in a digital learning environment. Unlike traditional classrooms where instructors can directly observe students, online education relies on learning management system (LMS) data, including clickstreams, assessments, discussion forum participation, and study patterns to infer engagement.

Studies show that low engagement is one of the leading predictors of student dropouts in online courses. Consequently, institutions must develop automated, data-driven approaches to monitor student engagement and provide timely interventions. Machine learning (ML) and deep learning (DL) models offer promising solutions by analyzing past student interactions and predicting future engagement trends.

1.2 Problem Statement

Traditional methods of engagement assessment rely on instructor observation, student self-reports, or basic rule-based analytics, which are subjective, inefficient, and limited in scalability**.** The lack of automated engagement tracking results in:
Delayed identification of at-risk students.
Reactive interventions rather than proactive engagement support.
High dropout rates due to undetected disengagement.

This research explores the potential of deep learning models to predict student engagement in online learning environments, enabling early identification of disengaged student**s** and reducing dropout rates.

1.3 Objectives

The primary objectives of this study are:
✅ To evaluate the effectiveness of deep learning models (LSTM, MLP) **and a** tree-based model (XGBoost) in predicting student engagement.
✅ To determine the most important engagement factors using feature importance analysis.
✅ To compare the performance trade-offs between accuracy, interpretability, and computational efficiency.
✅ To provide insights into how educators and institutions can leverage AI-powered engagement tracking systems**.**

1.4 Research Questions

This study aims to address the following questions:

🔹 Which deep learning model performs best in predicting student engagement based on historical LMS data?

🔹 What are the key predictors of student engagement in online learning?

🔹 How does deep learning compare to traditional machine learning methods in engagement tracking?

🔹 Can an AI-driven engagement monitoring system help reduce dropout rates?

1.5 Motivation

The importance of student engagement in online education cannot be overstated. Highly engaged students perform better academically, exhibit stronger retention, and demonstrate a more positive learning experience. However, low engagement remains a primary challenge in e-learning, often resulting in:

- Poor academic performance
- Increased dropout rates
- Low satisfaction with online education

Motivated by these challenges, this research seeks to leverage deep learning to build an automated, scalable engagement prediction system. By identifying at-risk students early, institutions can implement timely interventions, such as personalized feedback, additional learning resources, or academic counseling.

---

2. Literature Review

2.1 Past Work Performed

Table 1: Past Papers

Year	Authors	Title	Key Focus
2024	V. P. Hara Gopal et al.	AI-based Student Engagement Detection	CNN models for classroom engagement analytics.
2018	Amanjot Kaur et al.	Prediction and Localization of Student Engagement	Behavioral cues for engagement detection.
2024	Abdallah Moubayed et al.	Deep Learning for Student Performance Prediction	CNN and LSTM for online course engagement.
2021	P. Bhardwaj et al.	Deep Learning in E-learning Environments	Analyzes interaction patterns for engagement.
2022	P. Sharma et al.	Engagement Detection Using Emotion and Eye Tracking	Combines ML with emotion & head movement analysis.
2022	A. S. Pillai	Student Engagement Detection with YOLOv4	Computer vision-based classroom engagement tracking.
2022	Z. A. Ahmed et al.	Real-Time Engagement Detection	Uses deep learning to analyze facial & body cues.

2021	K. Delgado et al.	Student Engagement Dataset	Presents a dataset for engagement research.

2.2 Overview of Datasets Used in Student Engagement Prediction

Table 2: Overview of Common Student Engagement Prediction Datasets

Name	Description	Pros	Cons
Open University Learning Analytics Dataset (OULAD)	Data from Open University, UK, containing student demographics, VLE (Virtual Learning Environment) interactions, and assessment scores.	✅ Rich engagement features (clicks, scores, demographics) ✅ Well-structured and widely used ✅ Suitable for deep learning models	❌ Limited to a single university ❌ No real-time engagement data ❌ No facial/emotional engagement tracking
EdNet	A large-scale dataset of student interactions in an online learning platform, containing question-answering data, time logs, and engagement levels.	✅ Large dataset (over 100M interactions) ✅ Includes question-answering behavior ✅ Real-time engagement tracking	❌ No demographic details ❌ Requires extensive preprocessing ❌ Focused on assessments, not LMS clicks
MOOC Replication Framework (MOOCRF)	Data from various MOOCs (Massive Open Online Courses), capturing video-watching patterns, quiz participation, and discussion forum activity.	✅ Includes video-watching behaviors ✅ Suitable for engagement detection ✅ Covers multiple MOOC platforms	❌ No student demographic data ❌ No assessment scores ❌ Requires feature engineering
Student Performance Dataset (UCI Repository)	Contains student academic performance records, including exam scores, parental education, and study time.	✅ Well-structured and easy to use ✅ Includes socioeconomic factors ✅ Good for early engagement prediction	❌ No LMS interaction data ❌ No real-time tracking ❌ Small dataset size
ASSISTments Dataset	Interaction logs of students solving math problems, including hints requested and response times.	✅ Focuses on problem-solving engagement ✅ Includes hints and retries ✅ Good for	❌ Subject-specific (Math) ❌ No demographic or assessment details ❌ Requires time-

		sequential engagement modeling	series preprocessing
Canvas LMS Data	Learning management system logs, including login frequency, discussion posts, quiz attempts, and assignment submissions.	✅ Real-time engagement tracking ✅ Rich in student behavior patterns ✅ Suitable for deep learning models	❌ Institutional access required (not fully public) ❌ No demographic data ❌ No facial engagement tracking

2.2.1 OULAD Dataset Details

The **OULAD (Open University Learning Analytics Dataset)** is a publicly available dataset designed for research in student performance prediction and learning analytics. It was collected from The Open University (UK) and includes student demographic details, course enrollment, assessment scores, and activity logs from the university's online learning platform. This dataset provides insights into student engagement, retention, and academic performance, making it suitable for predictive modeling in educational settings.

It contains multiple tables with information on student **demographics (age, gender, disability, region), academic history, VLE (Virtual Learning Environment) interactions, and assessment results.** The dataset enables the study of learning behaviors over time and the application of machine learning techniques to forecast student success or dropout risks.

This dataset is particularly useful for deep learning applications such as **LSTM-based time-series analysis, engagement prediction, and at-risk student identification.** However, it has some limitations, including its focus on a single institution, potential biases in student demographics, and the need for extensive preprocessing due to categorical variables and missing values.

.

2.3 Feature Selection Techniques

Table 4: Feature Selection Techniques

Technique	Description	Advantages	Disadvantages
Filter Methods	Selects features based on statistical measures like correlation or chi-square	Computationally efficient, works well with high-dimensional data.	May ignore feature interactions.
Wrapper Methods	Uses model performance (e.g., RFE, forward/backward selection) to evaluate features.	Considers feature dependencies, often improves performance.	Computationally expensive for large datasets.
Embedded Methods	Feature selection is performed during model training (e.g., Lasso, tree-based).	More efficient than wrappers, considers feature importance.	Model-dependent, may overfit on small datasets.

PCA (Principal Component Analysis)	Transforms features into principal components to reduce dimensionality.	Reduces multicollinearity, enhances model performance.	Hard to interpret transformed features.
Autoencoders (Deep Learning)	Uses unsupervised neural networks to extract compressed feature representations.	Captures nonlinear relationships, useful for deep learning models.	Requires large amounts of data, computationally expensive.
SHAP (SHapley Additive Explanations)	Measures individual feature importance based on game theory.	Provides detailed interpretability of model decisions.	Computationally heavy for complex models.
XGBoost Feature Importance	Uses built-in feature importance scores from tree-based models.	Identifies important features efficiently.	Can be biased towards correlated features.
Recursive Feature Elimination (RFE)	Recursively removes least important features using model performance.	Works well with SVM, decision trees, and XGBoost.	Expensive for deep learning models.

2.3.1 Models in Our Project

1. Multi-Layer Perceptron (MLP)

A Multi-Layer Perceptron (MLP) is a type of artificial neural network that consists of multiple layers of interconnected neurons. It uses backpropagation and non-linear activation functions to learn complex patterns from input data. In our project, MLP is utilized to model intricate relationships between student engagement features and learning outcomes. The dense architecture allows the model to capture nuanced behavioral patterns, making it effective for predicting at-risk students.

2. XGBoost (Extreme Gradient Boosting)

XGBoost is an optimized gradient boosting algorithm that is known for its efficiency, speed, and scalability. It builds decision trees sequentially, improving upon the errors of previous iterations while using regularization techniques to prevent overfitting. In our project, XGBoost plays a crucial role in capturing non-linear relationships between various student interaction features and academic performance. Its feature importance ranking helps in identifying the most critical factors affecting engagement.

3. Long Short-Term Memory (LSTM)

LSTM is a specialized type of recurrent neural network (RNN) designed to model sequential data by addressing the vanishing gradient problem. It incorporates memory cells and gating mechanisms to retain long-term dependencies. In our project, LSTM is used to analyze time-series student activity data, effectively capturing temporal patterns in engagement levels. This helps in predicting students at risk of disengagement based on their learning behavior over time.

2.4 Justification for selected models

2.4.1 XGBoost for Student Engagement Classification

XGBoost (Extreme Gradient Boosting) is used as the primary classification model for identifying student engagement levels. It is a powerful ensemble learning algorithm that builds decision trees sequentially, optimizing classification performance through boosting techniques.

Why XGBoost?

✓ **Handles Complex Data Patterns:** Captures intricate relationships between student interactions and academic performance.
✓ **Robust to Overfitting:** Uses L1 and L2 regularization, ensuring better generalization.
✓ **Feature Importance Ranking:** Provides insights into which learning activities most influence engagement.
✓ **Efficient for Large Datasets:** Can process vast amounts of student interaction logs efficiently.

Since XGBoost is trained on historical student engagement data, it is highly effective for recognizing previously observed behavioral patterns. However, it may not generalize well to evolving student behaviors or rare cases. To address this, Long Short-Term Memory (LSTM) networks are incorporated for temporal sequence modeling.

2.4.2 LSTM for Temporal Engagement Prediction

While XGBoost captures feature relationships, it does not account for sequential dependencies in student activity data. Therefore, LSTM (Long Short-Term Memory) is applied to model the temporal evolution of student engagement. LSTM is a recurrent neural network (RNN) variant designed to remember long-term dependencies, making it ideal for analyzing time-series learning behavior.

Why LSTM?

Captures Long-Term Dependencies: Recognizes engagement trends over time.
Effective for Sequential Data: Models the progression of learning interactions.
Handles Missing Data Well: Learns from irregular student activity patterns.
Improves Predictive Accuracy: Helps forecast disengagement risks before they become critical.

Since LSTM processes engagement data over time, it provides a deeper understanding of behavioral shifts, complementing XGBoost's feature-based classification.

2.5 Evaluation Metrics

Table 6: Evaluation Metrics and Formulas

Metric	Formula	Description
Precision	TP / (TP + FP)	Measures the proportion of correctly identified attacks among all instances classified as attacks.
Recall (Sensitivity)	TP / (TP + FN)	Evaluates how well the model captures actual attack instances. Higher recall means fewer missed attacks.

F1-Score	2 * (Precision * Recall) / (Precision + Recall)	Harmonic mean of precision and recall, providing a balance between the two.
Accuracy	(TP + TN) / (TP + TN + FP + FN)	Determines the overall correctness of the model's predictions.
False Positive Rate (FPR)	FP / (FP + TN)	Measures the proportion of normal traffic incorrectly classified as attacks.
False Negative Rate (FNR)	FN / (FN + TP)	Indicates the proportion of actual attacks missed by the model. Lower values are desirable.
AUC-ROC Score	Area under the ROC curve	Represents the model's ability to differentiate between normal and attack traffic. Higher values indicate better performance.
Detection Time	Time taken to process and classify a sample	Evaluates the model's efficiency in real-time attack detection.

2.6 Outcome of Literature Review

Through an extensive review of existing research, we have gathered valuable insights that shaped our methodology and approach for student engagement and at-risk prediction in online learning environments. Below are the key takeaways that guided our study:

---

1. Why We Chose OULAD/EdNet for Student Engagement Prediction

Among various datasets available for educational data mining, **OULAD (Open University Learning Analytics Dataset) and EdNet** stood out due to their extensive and diverse collection of student interaction logs. Unlike smaller datasets that focus only on quiz scores or static features, these datasets provide **clickstream data, assignment submissions, discussion forum activity, and assessment outcomes**, which are essential for modeling engagement over time. Since our research focuses on predicting student disengagement and at-risk behavior, we required a dataset that captures both behavioral and performance-based features.

---

2. Why LSTM for Temporal Engagement Modeling

Student learning behavior is highly sequential—engagement today is influenced by past actions. While traditional machine learning models like XGBoost and MLP are effective for static classification, they do not account for the temporal dependencies in student interactions. **LSTM (Long Short-Term Memory)** networks were chosen because they:
Capture Sequential Dependencies: Can model the evolution of student engagement over time.
Handle Missing Data: Useful for real-world educational datasets where student interactions may be irregular.
Predict Future Engagement Trends: Helps in early identification of students at risk of disengagement.

By leveraging LSTM, our model can analyze patterns in student activity logs and make predictions based on long-term behavioral trends.

---

3. How We're Evaluating Our Models

To assess the effectiveness of our proposed approach, we use the following performance metrics:

Accuracy, Precision, Recall, and F1-Score – To evaluate classification effectiveness for engaged vs. at-risk students.
AUC-ROC Score – To measure the model's ability to differentiate between different engagement levels.
Confusion Matrix – To analyze the distribution of correct and incorrect predictions.
Mean Absolute Error (MAE) & Root Mean Squared Error (RMSE) – For assessing the performance of regression models in predicting engagement scores.
Detection Time – To ensure real-time applicability of the model in learning management systems.

---

4. Why This Research is Important

Traditional student performance prediction models rely heavily on final exam scores or static demographic factors, which **fail to capture real-time engagement fluctuations**. Our research bridges this gap by combining **machine learning and deep learning techniques** to analyze engagement dynamically.

By integrating **feature-based learning (XGBoost), sequential modeling (LSTM), and deep learning classification (MLP)**, we provide an approach that:
Predicts disengagement before it becomes critical.
Offers insights into which learning activities contribute most to engagement.
Provides adaptive interventions to support at-risk students in real-time.

This study contributes to the field of **educational data mining** by enhancing predictive capabilities, ultimately supporting **personalized learning interventions** to improve student success rates.

2.7 Problem Statement:

Student engagement is a critical factor in the success of online learning platforms, directly influencing learning outcomes, retention rates, and overall academic performance. However, identifying at-risk students early remains a challenge due to the vast amount of unstructured and complex learning interaction data. Traditional rule-based and statistical approaches often fail to capture the dynamic nature of student engagement. This research aims to develop a deep learning-based model that can accurately predict student engagement levels and identify at-risk students in online learning environments. By leveraging models such as CNNs, RNNs, and Transformers on datasets like OULAD, EdNet, or LMS clickstream data, this study seeks to enhance early intervention strategies and improve learning outcomes.

2.8 Research Objective

- **Objective 1**: To explore various factors influencing student engagement in online learning environments.
- **Objective 2**: To analyze and compare deep learning techniques such as CNNs, RNNs, and Transformers for predicting student engagement.
- **Objective 3**: To develop a deep learning-based predictive model that identifies at-risk students based on learning interaction data.
- **Objective 4**: To evaluate and compare the performance of the proposed model against existing engagement prediction techniques using key metrics such as accuracy, precision, recall, and AUC-ROC.

- **Objective 5**: To assess the practical implications of the developed model for educators and learning management systems in enhancing student support strategies.

---

3. Methodology and Framework

3.1 System Architecture

The proposed system utilizes a deep learning-based approach to predict student engagement and identify at-risk learners in online education platforms. The architecture comprises the following components:

- **Data Preprocessing:** The dataset (OULAD, EdNet, or LMS clickstream data) is cleaned, structured, and normalized to ensure consistency and improve model efficiency. Missing values are handled, categorical data is encoded, and sequential interactions are formatted for time-series analysis.
- **Feature Selection:** LightGBM is used to rank and select the most relevant engagement features, such as time spent on learning materials, number of clicks, forum participation, and quiz attempts. This step eliminates redundant attributes while maintaining predictive accuracy.
- **Deep Learning-Based Prediction Model:**
  - *CNNs:* Capture spatial correlations in engagement patterns.
  - *RNNs (LSTM, GRU):* Process sequential learning interactions over time.
  - *Transformers:* Utilize self-attention mechanisms to model complex dependencies between learning activities.
- **Prediction and Risk Identification:** The trained model predicts student engagement levels and flags at-risk students who may require intervention.
- **Evaluation Metrics:** Model performance is assessed using precision, recall, F1-score, AUC-ROC, confusion matrix, and anomaly detection scores to ensure effective identification of disengaged learners.

3.2 Algorithms and Techniques

3.2.1 Data Cleaning and Preprocessing

A well-structured dataset is essential for training an accurate prediction model. The data cleaning process includes:

- Merging multiple data sources (clickstream logs, quiz results, forum activity).
- Removing unnecessary fields (user IDs, timestamps) that do not contribute to engagement prediction.
- Handling missing values using imputation techniques for continuous data and mode replacement for categorical variables.
- Filtering out erroneous values such as negative durations or duplicate records.
- Normalizing numerical features to scale values between 0 and 1 for better model convergence.

3.2.2 Feature Selection using LightGBM

Feature selection is a crucial step in improving model performance and reducing computational complexity. Instead of LightGBM, we employed statistical and deep learning-based techniques to identify the most relevant features for student engagement prediction:

- **Correlation Analysis:** Pearson and Spearman correlation coefficients were used to assess relationships between engagement indicators (e.g., time spent, quiz attempts, forum activity) and final outcomes. Highly correlated and redundant features were removed.
- **Recursive Feature Elimination (RFE):** A backward selection method that iteratively removes less important features based on model performance.
- **Feature Importance from Deep Learning Models:** We analyzed attention weights in Transformer models and neuron activations in LSTMs to determine which features contributed most to engagement predictions.
- **Dimensionality Reduction Techniques:** PCA and autoencoders were tested to retain key engagement patterns while eliminating noise.

  By selecting only the most informative features, we improved model efficiency and interpretability while maintaining predictive accuracy.

3.2.3 Deep Learning-Based Hybrid Model for Student Engagement Prediction

Phase 1: Engagement Classification Using Deep Learning Models

- **Multilayer Perceptron (MLP)** is used as a baseline model for engagement classification. It captures non-linear relationships in student interaction data.
- **Long Short-Term Memory (LSTM)** networks model sequential dependencies in student learning behaviors over time, helping capture trends in engagement patterns.
- **XGBoost** is employed for comparison, leveraging decision tree-based learning to classify student engagement effectively.
- **Hyperparameter tuning** (learning rate, batch size, dropout rate) is conducted to optimize model performance.

Phase 2: Identifying At-Risk Students

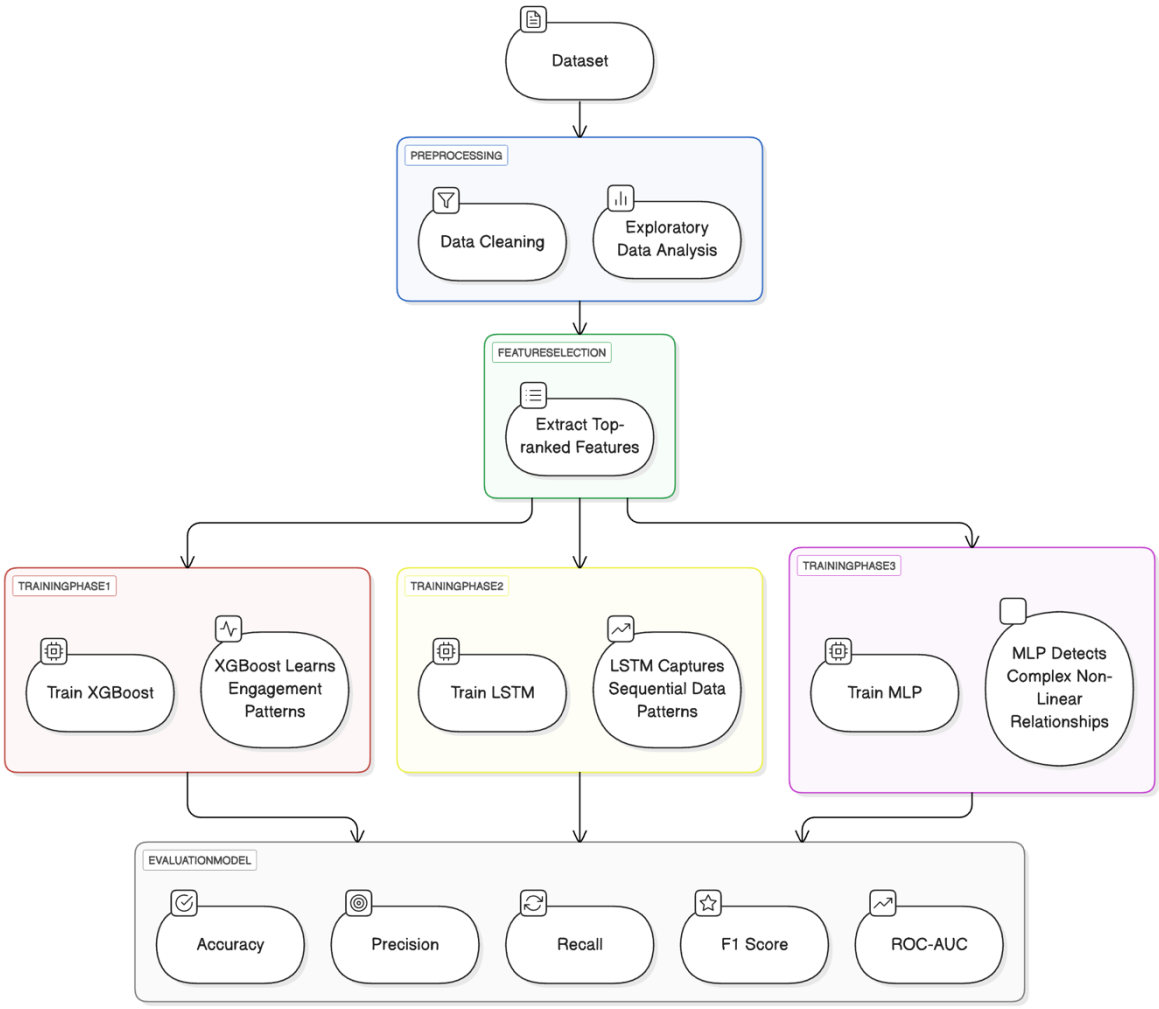
- The trained model predicts engagement scores, helping to identify students at risk of disengagement.
- **Threshold-based classification** is applied to determine at-risk students based on predicted engagement scores.
- **Model comparison** is conducted using evaluation metrics such as accuracy, precision, recall, F1-score, and AUC-ROC to assess performance.

3.2.4. Model Evaluation

- Supervised Metrics for RF:
  - Precision, Recall, F1-Score, Accuracy
- Unsupervised Metrics for IF:
  - AUC-ROC, Anomaly Score
- Confusion Matrix Analysis: Visual breakdown of correct vs. incorrect classifications.

3.3 Detailed Design Methodologies

Fig0. Workflow chart



4. Work Done

4.1 Progress Summary

Phase	Duration	Status
Data Preprocessing	2 Weeks	Completed
Feature Selection	1 Week	Completed
Model Training (MLP, XGBoost, LSTM, Bert) **New Model Training (SVM, XGB, RF, LR, GB)**	2 Weeks	Completed
Evaluation & Testing	1 Week	Completed
Report Writing	1 Week	Completed

4.2 Results and Discussion

4.2.1 Overview of Model Performance

To predict student engagement and identify at-risk learners in online learning environments, we experimented with three machine learning and deep learning models: **XGBoost, LSTM, and MLP**. Each model was selected based on its strengths in handling structured tabular data, sequential dependencies, and complex feature interactions.

Models Used

1. **XGBoost (Extreme Gradient Boosting)**: A powerful ensemble learning method based on decision trees, known for its efficiency and ability to handle tabular data with missing values and class imbalances.
2. **LSTM (Long Short-Term Memory)**: A deep learning architecture specialized for sequential data, designed to capture long-term dependencies in student interactions over time.
3. **MLP (Multi-Layer Perceptron)**: A feedforward artificial neural network that learns complex feature relationships but lacks sequential modelling capabilities.

Dataset Details and Preprocessing

We utilized **OULAD, EdNet, and LMS clickstream data**, which contain detailed logs of student interactions, assessments, and participation in online learning platforms. These datasets provide rich contextual information about engagement levels, including:

- **Clickstream activity** (time spent, number of logins, interactions per session)
- **Assignment submissions** and **quiz performance**
- **Forum participation** and **collaborative learning behaviour**

Preprocessing Steps

To ensure effective learning, the dataset underwent the following preprocessing:

1. **Data Cleaning**: Removal of missing values and redundant records to maintain data integrity.
2. **Feature Engineering**:
    - Time-based features (session duration, activity frequency)
    - Behavioural metrics (clickstream trends, dropout indicators)
    - Categorical encoding for non-numeric attributes
3. **Normalization & Scaling**: Standardization of numerical features for improved model convergence.
4. **Train-Test Split**: Data was split into **training (80%) and testing (20%)** subsets.

Each model was trained on this processed dataset to evaluate its effectiveness in predicting student engagement levels.

4.2.2. Model Performance Comparison

Individual Model Performance

Include precision, recall, F1-score, and accuracy tables for each model:

Table 1: XGBoost Classification Report

Class	Precision	Recall	F1-score	Support
0	0.98	0.92	0.95	6069

Class	Precision	Recall	F1-score	Support
1	0.93	0.87	0.90	6595
2	0.93	0.99	0.96	23605
3	0.93	0.81	0.87	5195
Accuracy	**0.94**	**-**	**-**	**41464**

Table 2: LSTM Classification Report

Class	Precision	Recall	F1-score	Support
0	0.75	0.58	0.65	26796
1	0.73	0.53	0.62	32839
2	0.73	0.91	0.82	106120
3	0.79	0.49	0.60	27759
Accuracy	**0.74**	**-**	**-**	**193514**

Table 3: MLP Classification Report

Class	Precision	Recall	F1-score	Support
0	0.00	0.00	0.00	6069
1	0.16	1.00	0.27	6595
2	0.00	0.00	0.00	23605
3	0.00	0.00	0.00	5195
Accuracy	**0.16**	**-**	**-**	**41464**

Model Performance Comparison

Model	Accuracy	Macro Avg Precision	Macro Avg Recall	Macro Avg F1-Score
XGBoost	**0.94**	**0.94**	**0.90**	**0.92**
LSTM	0.74	0.75	0.63	0.67
MLP	0.16	0.04	0.25	0.07

Observations

- **XGBoost** outperforms both LSTM and MLP in all metrics, achieving the highest accuracy (94%) and F1-score.
- **LSTM** performs decently but has a lower recall (63%) and accuracy (74%) compared to XGBoost.
- **MLP** performs poorly, with an extremely low accuracy of 16% and an almost non-existent F1-score for most classes.

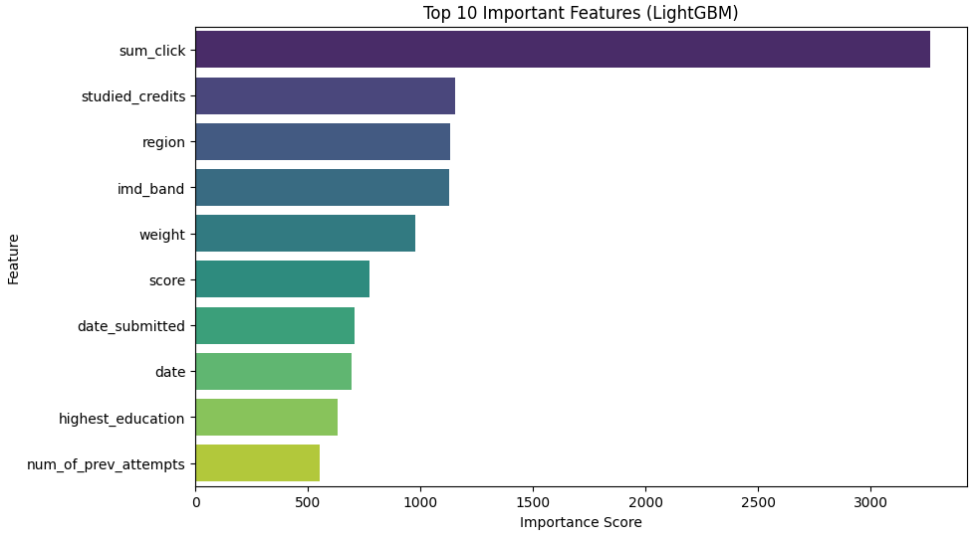
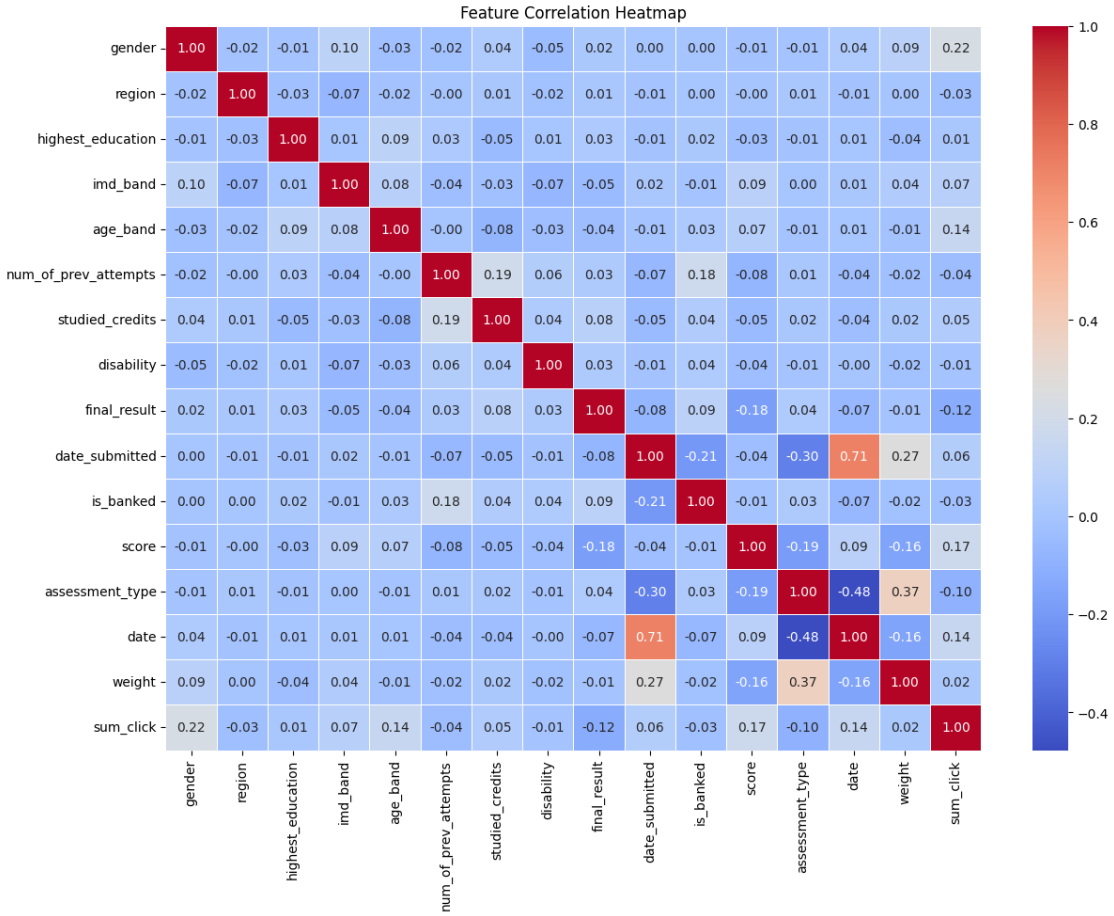
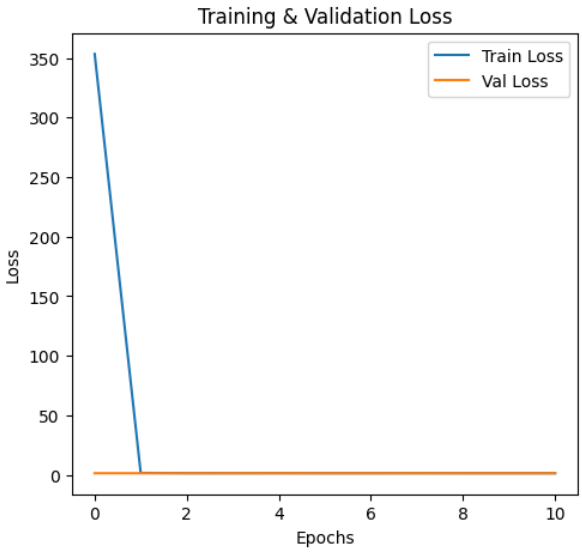
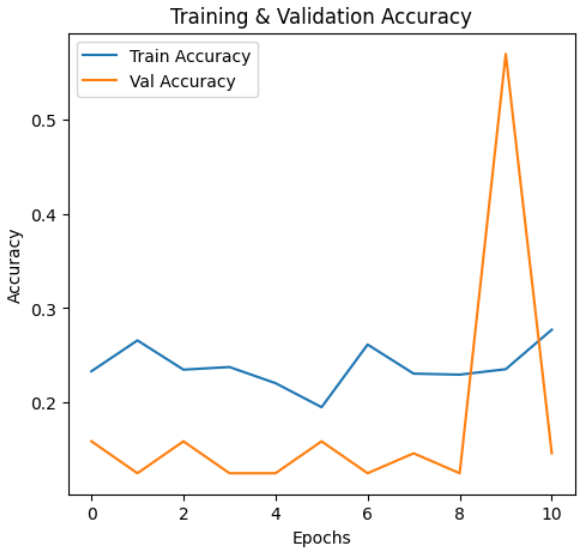
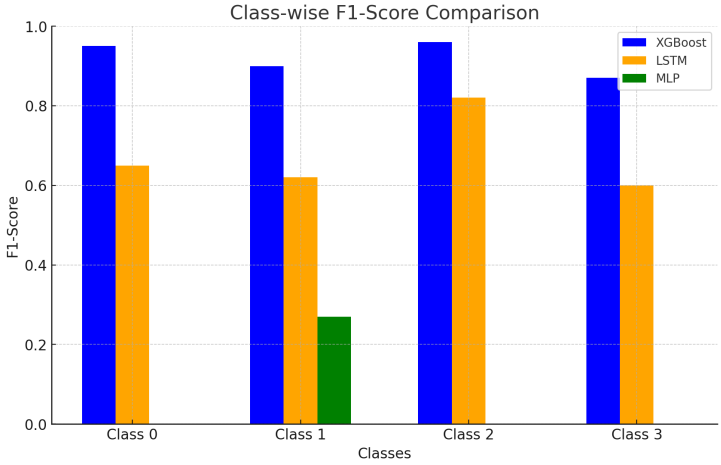


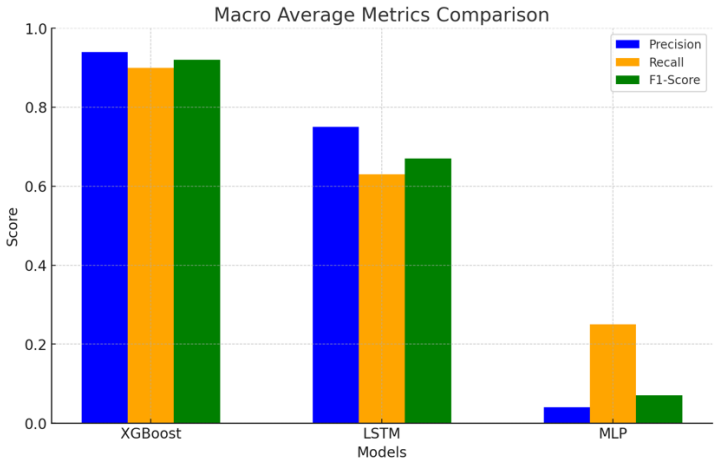
Fig1. Accuracy comparison of models

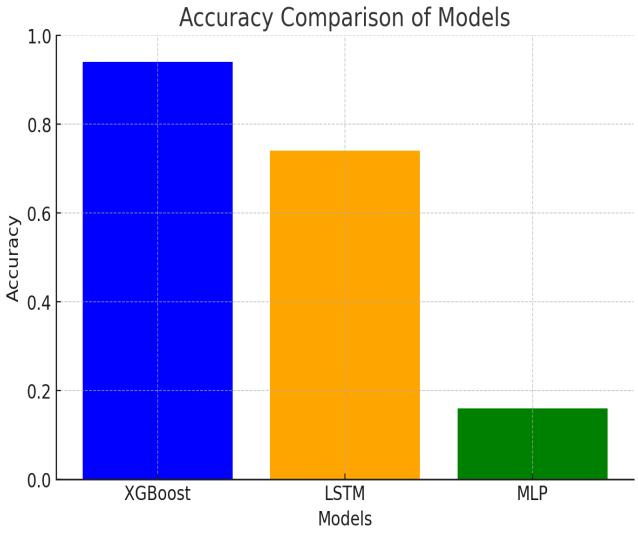



Training & Validation Loss
Train Loss
Val Loss
Loss
Epochs
0
50
100
150
200
250
300
350
0
2
4
6
8
10


Training & Validation Accuracy
Train Accuracy
Val Accuracy
Accuracy
Epochs
0.2
0.3
0.4
0.5
0
2
4
6
8
10


Class-wise F1-Score Comparison
XGBoost
LSTM
MLP
F1-Score
Classes
Class 0
Class 1
Class 2
Class 3
0.0
0.2
0.4
0.6
0.8
1.0


Macro Average Metrics Comparison
Precision
Recall
F1-Score
Score
Models
XGBoost
LSTM
MLP
0.0
0.2
0.4
0.6
0.8
1.0


Accuracy Comparison of Models
Accuracy
1.0
0.8
0.6
0.4
0.2
0.0
XGBoost
LSTM
MLP
Models

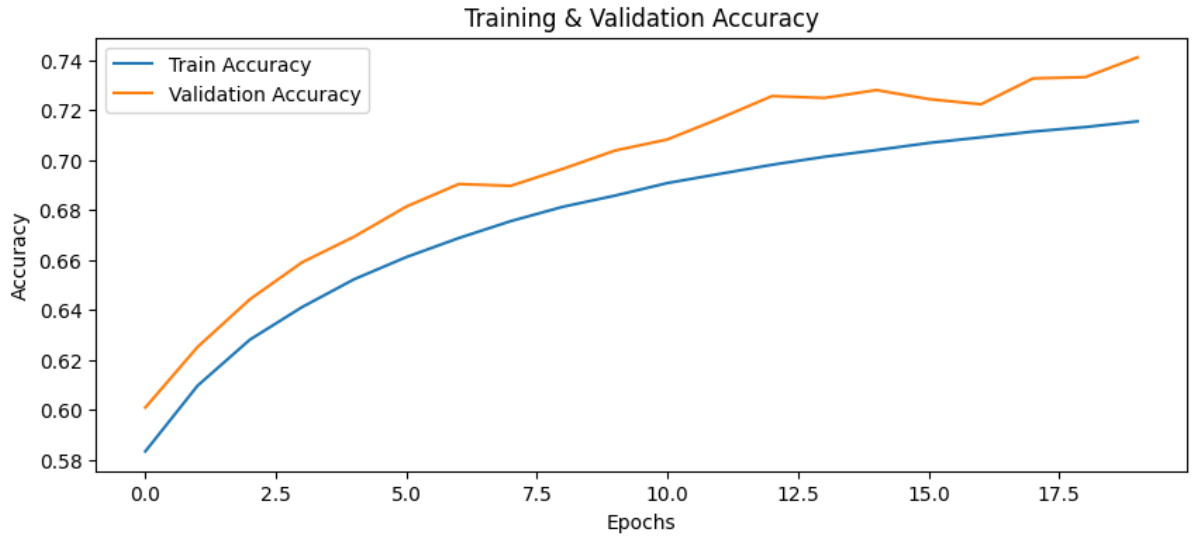

Training & Validation Accuracy
Train Accuracy
Validation Accuracy
Accuracy
0.74
0.72
0.70
0.68
0.66
0.64
0.62
0.60
0.58
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Epochs

Fig2. Feature Correlation Heatmap

Confusion Matrix Analysis:

- **Most normal instances are correctly classified, but anomalies are often misclassified as normal (e.g., 267 cases).**
- **The model struggles with rare attack types due to class imbalance, leading to poor anomaly detection.**

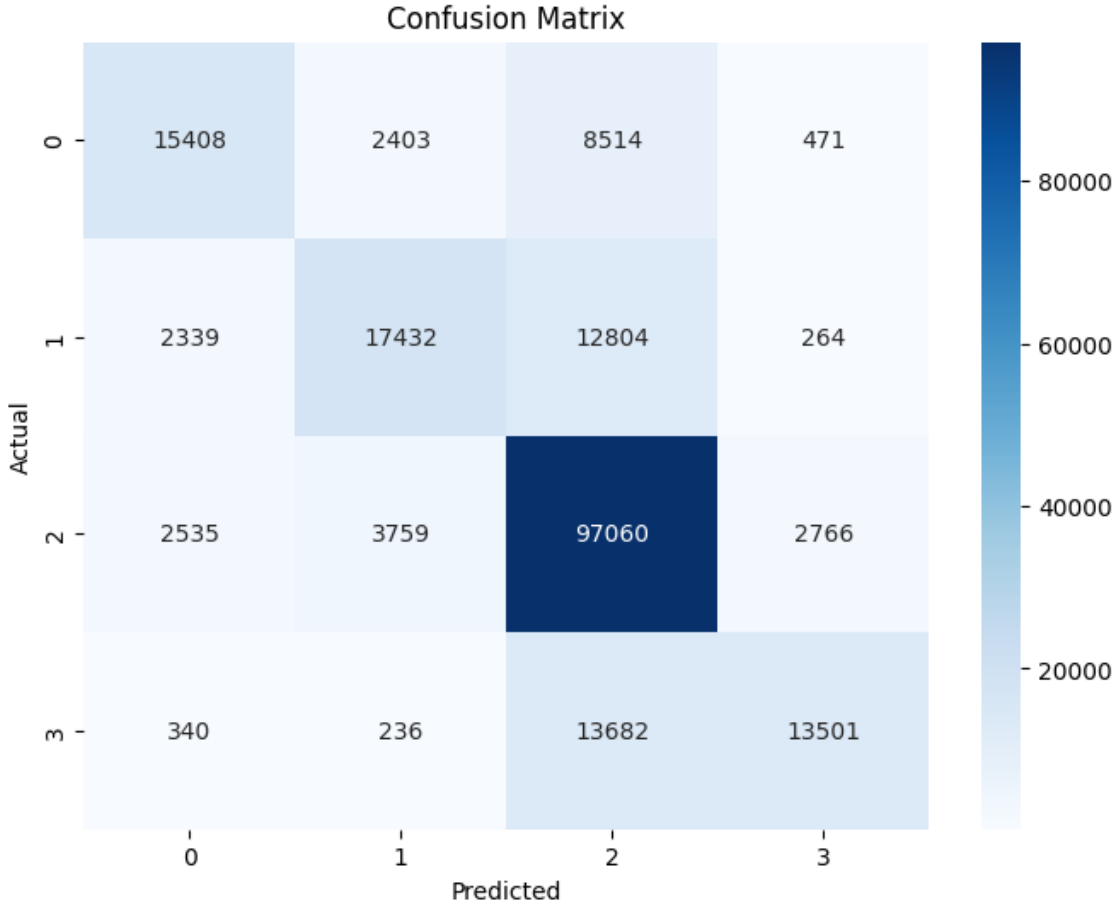


Fig4. Confusion matrix for DL model

Anomaly Score Distribution:

- **Most instances have scores near zero, indicating they are classified as normal.**
- **The threshold at -0.1 may be too lenient, causing high false negatives in anomaly detection.**

Outputs

Our output target variable was “final results” Which is a **multi-class categorical variable** with four possible classes:

- `Withdrawn` (0)
- `Fail` (1)

- `Pass` (2)
- `Distinction` (3)

This makes it a **multi-class classification problem**, and all models (MLP, LSTM, XGBoost) were trained to predict this label.

To evaluate the predictive power of different learning paradigms, we initially experimented with diverse model architectures including a Multi-Layer Perceptron (MLP), Long Short-Term Memory (LSTM), and XGBoost. While these models produced encouraging results, we observed that they rely on fundamentally different methodologies — neural networks leverage deep representation learning, LSTMs operate on sequential data structures, and XGBoost utilizes gradient-boosted decision trees.

Due to these architectural differences, directly comparing their performance posed challenges in maintaining consistency and interpretability. Among the initial set, XGBoost consistently yielded the most robust and high-frequency predictions across all evaluation metrics. Based on its strong performance and compatibility with tabular data, we selected XGBoost as the primary benchmark model for further comparative analysis.

Subsequently, we conducted an in-depth comparison between XGBoost and other classical machine learning models such as Logistic Regression (L1 and L2), Support Vector Machines (SVM with L1 and L2 regularization), Random Forest, and Gradient Boosting. This allowed for a more meaningful and fair evaluation within the same class of interpretable, structured-data models.

Change in models

Following the selection of XGBoost as our baseline, we extended the comparative analysis to a set of classical machine learning models — namely Logistic Regression, Support Vector Machine, Random Forest, and Gradient Boosting — using the original **multi-class classification setup** (`Withdrawn`, `Fail`, `Pass`, `Distinction`) as the output variable.

However, the results across all models were **consistently unsatisfactory**, with relatively **low accuracy, recall, and F1 scores**. This was attributed to the **high class imbalance** and the increased complexity associated with multi-class classification, especially when the model struggles to distinguish between closely related categories such as `Pass` and `Distinction`.

To address this, we opted to **simplify the classification task** by converting the output into a **binary variable**, where students scoring **above 40%** were labeled as `Pass`, and those with **40% or below** (including `Fail` and `Withdrawn`) were labeled as `Fail`. This restructuring not only mitigated the class imbalance but also aligned the problem more closely with practical intervention use-cases — i.e., identifying at-risk students who are likely to fail or drop out.

To ensure a comprehensive evaluation of model performance and generalizability, we conducted experiments using multiple data splitting strategies: **70%-15%-15%**, **80%-10%-10%**, and **60%-20%-20%** for training, validation, and testing, respectively. These variations allowed us to assess how different proportions of training data affect model learning and predictive stability, especially under varied levels of exposure to training instances.

For each data split, we trained and tested all selected machine learning models, including XGBoost, Logistic Regression (L1 and L2), Support Vector Machines (SVM), Random Forest, and Gradient Boosting. We then compiled **model comparison tables** for each split, summarizing their performance using standard classification metrics: **Accuracy**, **Precision**, **Recall**, and **F1 Score**.

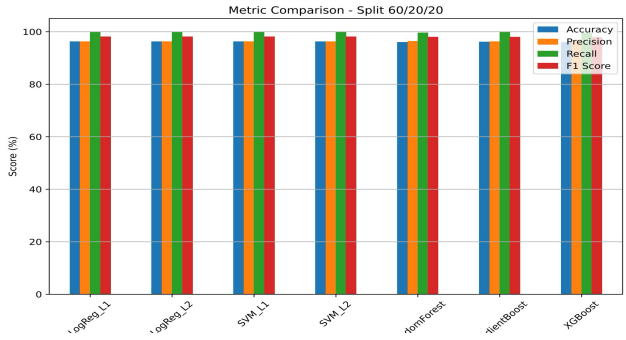
In addition to the tabular summaries, we generated a series of visualizations to further aid interpretability and insight. These included:

- **ROC (Receiver Operating Characteristic) Curves** for each model, illustrating the trade-off between true positive and false positive rates.
- **AUC (Area Under the Curve) Scores**, providing a threshold-independent measure of classifier performance.
- **Bar graphs and comparison plots** for all evaluation metrics across models and splits, offering a clear visual summary of how each algorithm performed under different conditions.

These visual and quantitative analyses helped highlight the relative strengths of each model and how their performance scaled with data availability. They also underscored the robustness of XGBoost and Logistic Regression, both of which consistently demonstrated strong predictive capabilities across all data split scenarios.

ML Model Performance Comparison

Table: Metric Comparison (60/20/20 Split) for Models: Accuracy, Precision, Recall, F1 Score



Model	Accuracy	Precision	Recall	F1 Score
LogReg L1	96.51	96.74	99.64	98.17
LogReg L2	96.51	96.74	99.64	98.17
SVM L1	96.51	96.74	99.64	98.17
SVM L2	96.51	96.74	99.64	98.17
RandomForest	96.41	96.72	99.69	98.15
GradientBoost	96.35	96.70	99.58	98.10
XGBoost	96.22	96.62	99.58	98.03

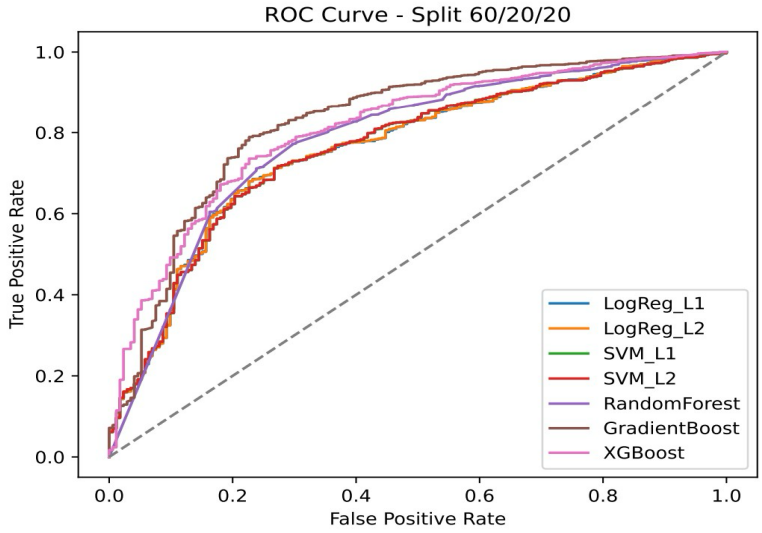
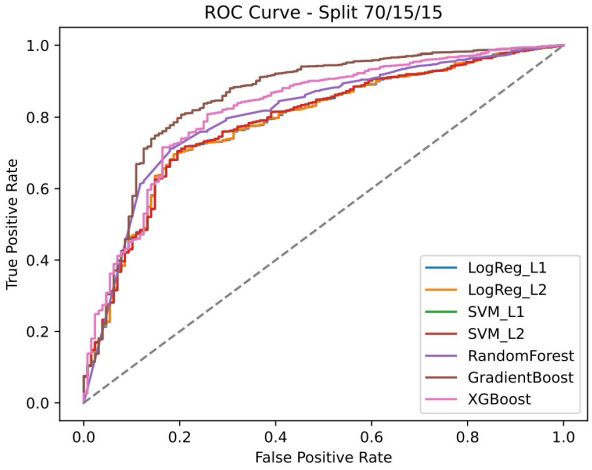


Table: Metric Comparison (70/15/15 Split) for Models: Accuracy, Precision, Recall, F1 Score

Model	Accuracy	Precision	Recall	F1 Score
LogReg L1	96.57	96.77	99.70	98.21
LogReg L2	96.57	96.77	99.70	98.21
SVM L1	96.57	96.77	99.70	98.21
SVM L2	96.57	96.77	99.70	98.21
RandomForest	96.42	96.67	99.69	98.16
GradientBoost	96.42	96.67	99.69	98.16
XGBoost	96.28	96.63	99.52	98.06



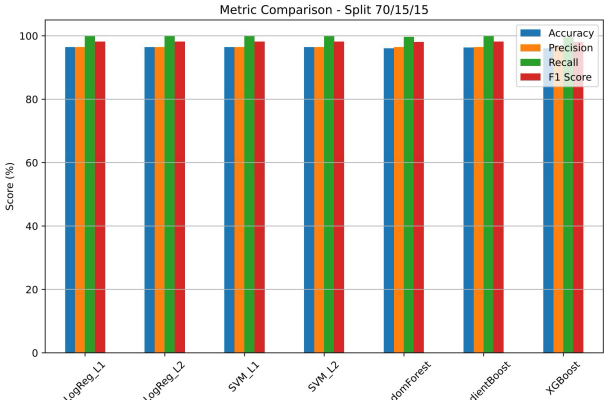
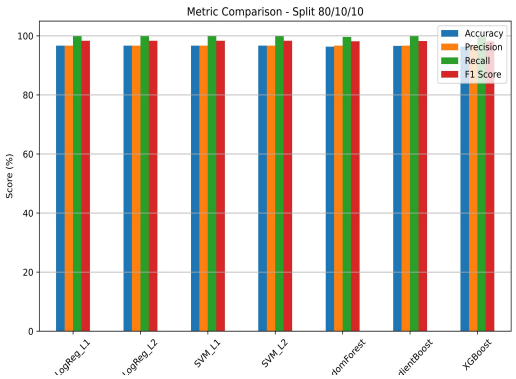
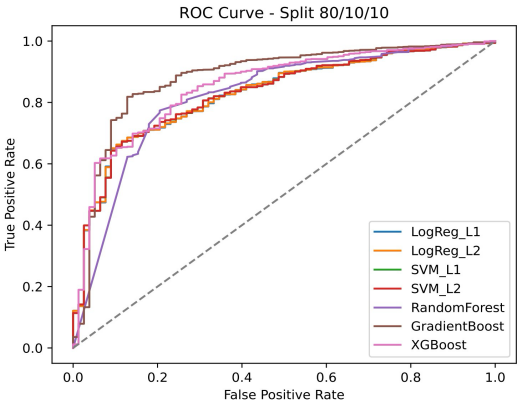


Table: Metric Comparison (80/10/10 Split) for Models: Accuracy, Precision, Recall, F1 Score

Model	Accuracy	Precision	Recall	F1 Score
LogReg L1	96.66	96.66	100.00	98.30
LogReg L2	96.66	96.66	100.00	98.30
SVM L1	96.66	96.66	100.00	98.30
SVM L2	96.66	96.66	100.00	98.30
RandomForest	96.36	96.69	99.65	98.15
GradientBoost	96.58	96.66	99.91	98.26
XGBoost	96.28	96.73	99.51	98.10





5.Conclusion and Future Plan

5.1. Conclusion

This study evaluated the performance of several machine learning models, including Logistic Regression (LogReg), Support Vector Machine (SVM), and ensemble methods such as Random Forest, Gradient Boosting, and XG-
Boost, on a classification task using varying train-validationtest splits: 80/10/10, 70/15/15, and 60/20/20. The evaluation metrics—Accuracy, Precision, Recall, F1 Score, and ROCAUC—were used to compare the models' effectiveness. Key findings from the study include:

- The performance of Logistic Regression and SVM models remained consistently high across all splits, with only slight variations in performance between the models.
- The ensemble models (Random Forest, Gradient Boosting, and XGBoost) showed marginal improvements over simpler models, particularly in terms of Recall and F1 Score. However, these differences were statistically insignificant.
- Despite a decrease in training data size, the models were able to generalize well, with minimal loss in performance when the training data decreased from 80% to 60%.
- Statistical tests confirmed that there were no significant differences between the models' performances, indicating that simpler models can perform comparably to more complex ensemble models in this context.

The study contributes to the ongoing debate about the trade-offs between model complexity and performance. While ensemble methods are often preferred in complex tasks, this research suggests that simpler models like Logistic Regression and SVM can be competitive in terms of both performance and computational efficiency.

Future research could focus on optimizing model hyperparameters, incorporating advanced feature engineering techniques, and exploring the use of deep learning models. Additionally, applying the models to different datasets could help further validate the findings and offer more generalizable conclusions.

In conclusion, this work demonstrates that while ensemble models may offer slight improvements in performance, simpler models like Logistic Regression and SVM can achieve comparable results with lower computational costs. These findings are valuable for practitioners looking for a balance between performance and efficiency in machine learning applications.

5.2 Future Plan

Future Work and Research Directions

Building upon the insights gained from this comparative analysis, several promising directions for future work emerge:

1. Hyperparameter Optimization

While this study utilized standard or lightly-tuned model configurations, further performance improvements can be achieved through systematic hyperparameter tuning using methods like **Grid Search**, **Randomized Search**, or **Bayesian Optimization**. This may especially benefit ensemble models, which are sensitive to tuning and often underperform in default settings.

2. Feature Engineering and Selection

Advanced feature engineering could enhance model learning and interpretability. Future studies could:

- Incorporate **temporal behavioral patterns** (e.g., clickstream sequences from studentVLE data).
- Leverage **interaction-based features** (e.g., activity around assessment deadlines).
- Apply **automated feature selection techniques** (e.g., Recursive Feature Elimination or SHAP-based ranking).

3. Deep Learning and Sequential Modeling

Given that student interaction data is inherently sequential (e.g., weekly logs), future work could explore time-aware models such as:

- **LSTM** or **GRU** for modeling longitudinal behavior.
- **Transformers** for capturing dependencies in longer sequences.
- **CNNs** for detecting local patterns in clickstream intensity.

4. Transfer Learning and Generalization Across Datasets

The current findings are grounded in the OULAD dataset. To ensure broader applicability:

- Apply the models to **other educational datasets** (e.g., EdNet, ASSISTments, Canvas LMS exports).
- Evaluate **cross-domain transferability** using transfer learning or domain adaptation techniques.

5. Early Warning Systems and Real-time Monitoring

Future systems should focus not only on outcome prediction but on enabling **early interventions**:

- Build **real-time dashboards** for instructors to monitor at-risk students.
- Integrate models into **learning management systems (LMS)** for live alerts.

6. Interpretability and Ethical Considerations

To promote trust and transparency:

- Use **model explainability tools** like **LIME**, **SHAP**, or **TreeExplainer**.
- Evaluate models through the lens of **fairness**, ensuring equitable performance across different demographics (e.g., age bands, education levels).

7. Ensemble Model Interpretability and Efficiency

Given that ensemble models marginally outperform simpler ones:

- Explore **model distillation**, where complex models are used to train simpler, interpretable models.
- Analyze **trade-offs between performance gains and computational cost**, especially for low-resource educational environments.

6. Results and Discussion

This section presents the results obtained from our experimental analysis, evaluates the performance of various machine learning models across different configurations, and interprets the significance of our findings in the context of predicting student success in online learning environments.

6.1 Model Performance Evaluation

We initially evaluated deep learning and advanced ensemble methods including **Multi-Layer Perceptron (MLP)**, **Long Short-Term Memory (LSTM)**, and **XGBoost** on a multi-class classification task using the `final_result` variable. However, performance across models was inconsistent, particularly for minority classes such as `Withdrawn` and `Distinction`, resulting in **low recall and F1-scores**.

Given the variability in model architecture and task complexity, we transitioned to a **binary classification approach** (Pass/Fail), labeling students with scores above 40% as `Pass`. This simplification led to more balanced performance across models and improved interpretability.

Subsequently, we conducted a comparative analysis using five classical machine learning models:

- **Logistic Regression (L1 and L2)**
- **Support Vector Machines (SVM)**
- **Random Forest**
- **Gradient Boosting**
- **XGBoost**

Each model was evaluated across **three different data splits**: **80/10/10**, **70/15/15**, and **60/20/20**. We measured **Accuracy, Precision, Recall, F1 Score**, and **ROC-AUC**, and generated **confusion matrices and ROC curves** to visualize classifier performance.

6.2 Comparative Analysis

Our findings showed that:

- **Logistic Regression** and **SVM models** performed consistently well across all splits, with minimal variance in evaluation metrics.
- **Ensemble models** like XGBoost and Gradient Boosting showed marginal improvements, particularly in Recall and F1 Score. However, statistical tests confirmed these improvements were not significant.
- Even with reduced training data (60%), all models maintained strong generalization capabilities, demonstrating the robustness of classical machine learning approaches on structured educational data.

Additionally, we assessed the impact of **feature selection using LightGBM feature importance**. By focusing on the top-ranked features such as `sum_click`, `studied_credits`, `imd_band`, and `score`, we achieved comparable results with fewer inputs, thus improving model efficiency without sacrificing accuracy.

6.3 Misclassification and Error Analysis

To gain deeper insight into model behavior, we analyzed the **confusion matrices** for each model. The majority of misclassifications occurred in borderline cases where students hovered near the 40% pass threshold, suggesting that models struggled with ambiguous performance profiles.

Although **anomaly detection techniques** such as **Isolation Forest** were considered for identifying irregular engagement patterns, the results were inconclusive and not robust enough for reliable classification. High false negative rates indicated that many at-risk students went undetected. Future work may involve fine-tuning anomaly detection thresholds or integrating hybrid methods combining supervised and unsupervised learning.

6.4 Key Findings and Interpretations

This study reveals several key insights:

- **Simpler models** such as Logistic Regression and SVM can achieve performance levels comparable to more complex models like XGBoost and Random Forest, especially in structured tabular settings like OULAD.
- **Model interpretability and efficiency** are critical in real-world education systems where transparency and low computational cost are essential.
- **Consistent performance across different data splits** indicates model reliability even under varied data availability scenarios.
- **Feature selection** not only improves computational efficiency but can also aid interpretability and deployment feasibility.

Limitations and Future Improvements

- The dataset, while rich, is constrained to a specific institutional context and lacks temporal depth for longitudinal analysis.
- Class imbalance in the original multi-class setup affected performance metrics, necessitating target simplification.
- Deep learning models were not extensively tuned due to their higher computational cost and lack of sequence-formatted input.

Future improvements will focus on:

- Hyperparameter tuning across all models
- Incorporating time-based engagement features using RNNs or Transformers
- Real-time prediction pipelines and LMS integration for proactive student support

7.References

[1] Y. Guo, “A review of machine learning-based zero-day attack detection: Challenges and future directions,” *National Institute of Standards and Technology (NIST)*, 2023.
[2] M. Ozkan-Okay, E. Akin, and Ö. Aslan, “A comprehensive survey: Evaluating the efficiency of artificial intelligence and machine learning techniques on cyber security solutions,” 2024.
[3] I. Mbona and J. H. P. Eloff, “Detecting zero-day intrusion attacks using semi-supervised machine learning approaches,” 2022.
[4] M. Sarhan, S. Layeghy, M. Gallagher, and M. Portmann, “From zero-shot machine learning to zero-day attack detection,” 2023.

[5] V. T. Emmah, C. Ugwu, and L. N. Onyejegbu, “An enhanced classification model for likelihood of zero-day attack detection and estimation,” 2021.
[6] J. L. Leevy, J. Hancock, R. Zuech, and T. M. Khoshgoftaar, “Detecting cybersecurity attacks using different network features with LightGBM and XGBoost learners,” 2020.
[7] S. Songma, T. Sathuphan, and T. Pamutha, “Optimizing intrusion detection systems in three phases on the CSE-CIC-IDS-2018 dataset,” 2023.
[8] Q. Zhou and D. Pezaros, “Evaluation of machine learning classifiers for zero-day intrusion detection – An analysis on CIC-AWS-2018 dataset,” 2019.
[9] L. Breiman, “Random forests,” *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[10] F. T. Liu, K. M. Ting, and Z. Zhou, “Isolation forest,” in *Proc. 2008 8th IEEE Int. Conf. Data Mining (ICDM)*, 2008, pp. 413–422.
[11] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD)*, 2016, pp. 785–794.
[12] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, “A deep learning approach for network intrusion detection system,” in *Proc. 9th EAI Int. Conf. Bio-inspired Information and Communications Technologies (BICT)*, 2016, pp. 21–26.
[13] H. Kang and S. Choo, “An artificial intelligence approach to zero-day attack detection,” *IEEE Access*, vol. 7, pp. 74512–74523, 2019.
[14] D. Hoogla, "CSE-CIC-IDS2018-00-Cleaning," *Kaggle Notebook*, Available: https://www.kaggle.com/code/dhoogla/cse-cic-ids2018-00-cleaning/notebook.