Reducing Dropout Rates in Online Learning PLatforms : A Predictive and Recommender System Using Machine Learning

Atmani Oumaima
Abdelmalek Essaadi University
oumaima.atmani@etu.uae.ac.ma

Supervised by professor KHAMJANE AZIZ
Abdelmalek Essaadi University
akhamjane@uae.ac.ma

1.Abstract

In this project we aim to adress the pressing issue of high dropout rates in online learning platforms . Using the OPEN UNIVERSITY LEARNING ANALYTICS DATASET (OULAD), we developed a machine learning-based predictive model to identify students at risk of not completing their courses. Additionally, a personalized recommender system was created to provide at-risk students with actionable interventions, including tailored resources and support. This integrated approach combines predictive analytics with targeted recommendations, offering a scalable solution for enhancing student engagement and retention in online learning environments.

2. INTRODUCTION

Massive Open Online Courses (MOOCs) allow people to take courses from a
variety of institutions and instructors via the internet anytime, anywhere. Despite the
growing popularity of systems such as Coursera , Open edX , and Udacity, they
are also plagued with the issue of low completion .
Student dropout rates are a persistent challenge in online learning platforms, negatively affecting both learners and institutions. The lack of personalized support and timely intervention often leads to disengagement and failure to complete courses.
This project aims to :

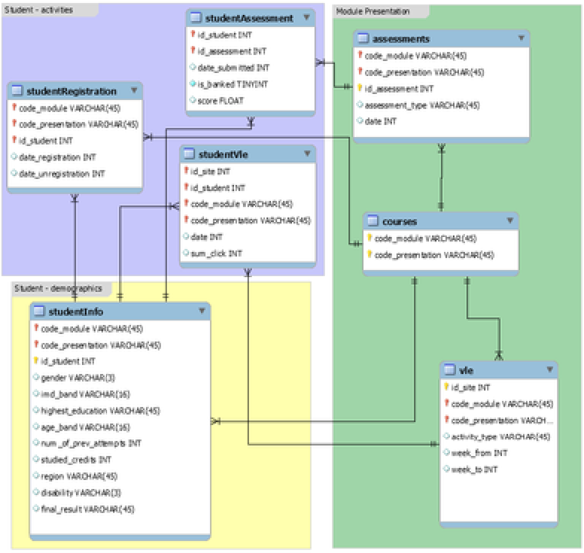
1. Build a predictive model to identify students at risk of not completing their courses.
2. Develop a recommender system to suggest personalized interventions .

Context:
We choosed o use the OULAD for its rich and diverse features representing student interactions , demographics and performance data while focused on a university context beside the dataset's structure and insights are highly relevant to general online learning platforms .

3.Methodology

3.1.Data Collection

the anonymised Open University Learning Analytics Dataset (OULAD) contains data about courses, students and their interactions with Virtual Learning Environment (VLE) for seven selected courses (called modules). Presentations of courses start in February and October - they are marked by "B" and "J" respectively. The dataset consists of tables connected using unique identifiers. All tables are stored in the csv format.



1. studentInfo.csv
This dataset contains information about students' demographics and socioeconomic status.
2. studentRegistration Dataset
This dataset contains registration details for students, such as course enrollments.
3. courses Dataset
Contains details about the courses.
4. assessments Dataset
Contains information about course assessments.
5. studentAssessment Dataset
Contains the results of student assessments.
6. studentVle Dataset
Contains information about student interactions with the Virtual Learning Environment (VLE).
7. vle Dataset
Contains information about the VLE sites.

3.2.Data preprocessing

To create a comprehensive dataset capturing all relevant information about students, courses, and interactions, we merged multiple data sources using common keys.

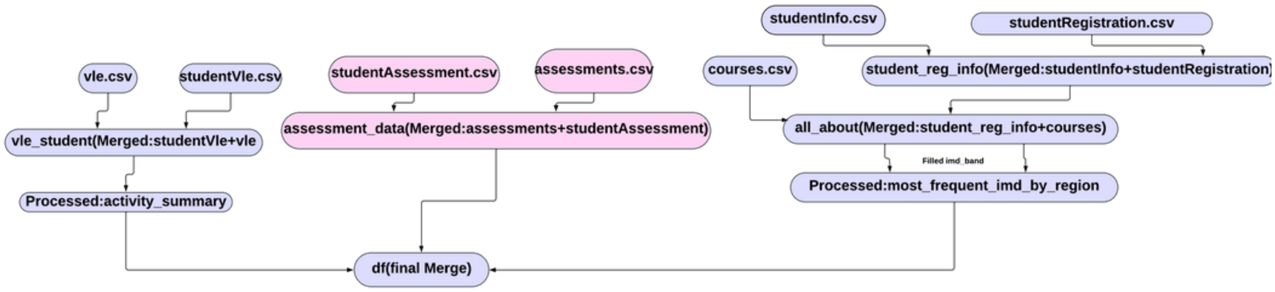

studentInfo.csv
studentRegistration.csv
vle.csv
studentVle.csv
studentAssessment.csv
assessments.csv
courses.csv
student_reg_info(Merged:studentInfo+studentRegistration)
vle_student(Merged:studentVle+vle
assessment_data(Merged:assessments+studentAssessment)
all_about(Merged:student_reg_info+courses)
Filled imd_band
Processed:activity_summary
Processed:most_frequent_imd_by_region
df(final Merge)

FIGURE 1. Data Merging and Integration Workflow

4. Exploratory Data Analysis (EDA) and feature enginnering

The exploratory data analysis provided key insights into student behavior and course performance. Key observations revealed that student engagement, timely submissions, and assessment scores are significant predictors of course completion. Based on these findings and the final merged dataset, we engineered the following features to enhance the predictive power of our model:

1. Temporal Engagement Features:
2. Engagement Metrics:
3. Assessment-Specific Features:
4. Engagement by Resource:
5. Behavioral and Performance Indicators:

5.Scaling and encoding

To prepare the data for modeling, preprocessing steps were applied to ensure compatibility with machine learning algorithms:

1. Scaling of Numerical Features:
   - Continuous variables were normalized using standard scaling to ensure a consistent range.
2. Encoding of Categorical Variables:
   - Categorical variables were one-hot encoded.
   - Binary variables were label-encoded for simplicity.

These transformations ensured that all features were in a machine-learning-ready format.

6.Modeling and analysis

6.1.Problem definition

With the preprocessed dataset finalized, the next step involved building and evaluating predictive models to identify students at risk of dropping out. This phase focuses on model selection, hyperparameter tuning, and performance evaluation.

This is a binary classification problem with the target variable completion_status having two classes:

- 0: Finished the course.
- 1: Did not finish the course.

Due to the class imbalance (more students completing than dropping out), we used SMOTE (Synthetic Minority Oversampling Technique) to balance the dataset.

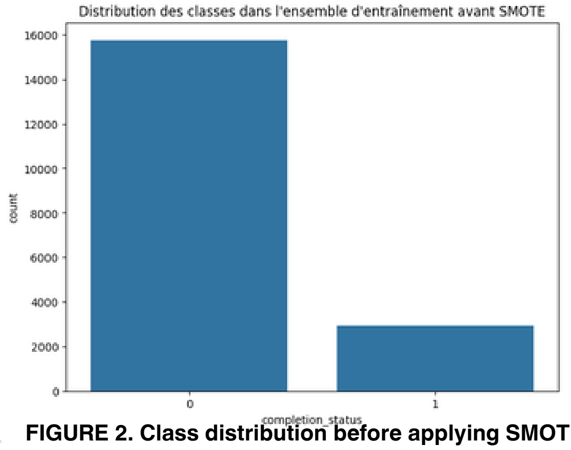


FIGURE 2. Class distribution before applying SMOTE

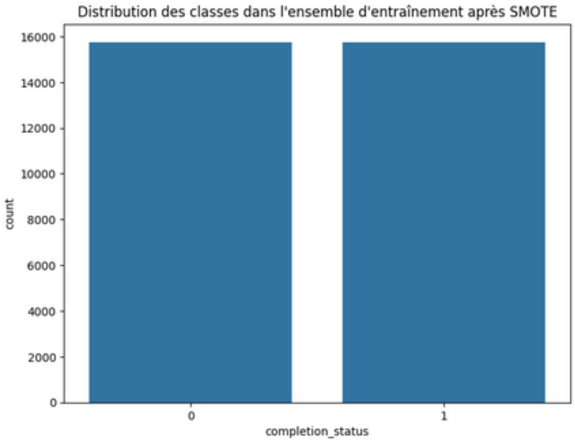


FIGURE 3. Class distribution after applying SMOTE

6.2.Models Used

We fit the following models to the training data, each tailored to capture different patterns in the data:

(a) Logistic Regression:

Logistic Regression models the probability of the target class as a sigmoid function of a linear combination of input features. The probability P(y=1 | x) is given by :

$$P(y=1|x) = \frac{1}{1+e^{-(w^T x+b)}}$$

W : weight vector

b:the bias term

We minimized the log-loss function with L2 regularization to prevent overfitting

$$\mathcal{L}(\theta) = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)] + \lambda\|\theta\|_2^2$$

(b) Decision Tree Classifier

We fit the decision tree classifiers that splits the data into subsets based on feature thresholds that maximize information gain or minimize Gini impurity:

$$Gini = 1 - \sum_{k=1}^{K} p_k^2$$

pk is the probability of class k at a given node. Each split aims to reduce the overall impurity of the tree.

(c) Random Forest Classifier

Random Forest is an ensemble learning algorithm that trains multiple decision trees on bootstrapped datasets and averages their predictions:

$$\hat{y} = \text{mode}\{h_t(x) \mid t = 1, \ldots, T\}$$

where ht(x) is the prediction of the t the tree.

(d) Support Vector Machine (SVM)

SVM constructs a hyperplane that maximizes the margin between classes. For a linear kernel:

$$f(x) = w^T x + b$$

(e) XGBoost Classifier

We also fit the XGBoost model that is a gradient-boosting algorithm that that iteratively builds decision trees to minimize the prediction error by focusing on the mistakes made by previous models.

$$Obj = \sum_{i=1}^{n} L(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$$

- L(yi,ŷi) is the binary log-loss, and the regularization term Ω(fk) penalizes tree complexity.

The objective function for XGBoost is designed to minimize the following loss function:

$$L(\hat{y}_i, y_i) = -\left(y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)\right)$$

  - yi represents the actual label (1 for "dropped out", 0 for "completed").
  - ŷi represents the predicted probability of dropout.

- The binary log-loss function penalizes the model when the predicted probability of dropout deviates significantly from the actual outcome.

(f) Multilayer Perceptron (MLP)

We also fit the MLP which is a feedforward neural network that consists of at least three layers with one or more hidden layers. Each neuron in a layer is connected to every neuron in the next layer, and these connections are associated with weights that are updated during training.

1. **Activation Function:**

In the hidden layers, the Rectified Linear Unit (ReLU) activation function is applied to introduce non-linearity:

$$h(x) = \max(0, Wx + b)$$

2.Output Layer:

The output layer uses the softmax function for multi-class problems or the logistic sigmoid function for binary classification. In this case, the sigmoid function was used:

$$\hat{y} = \frac{1}{1 + e^{-z}}$$

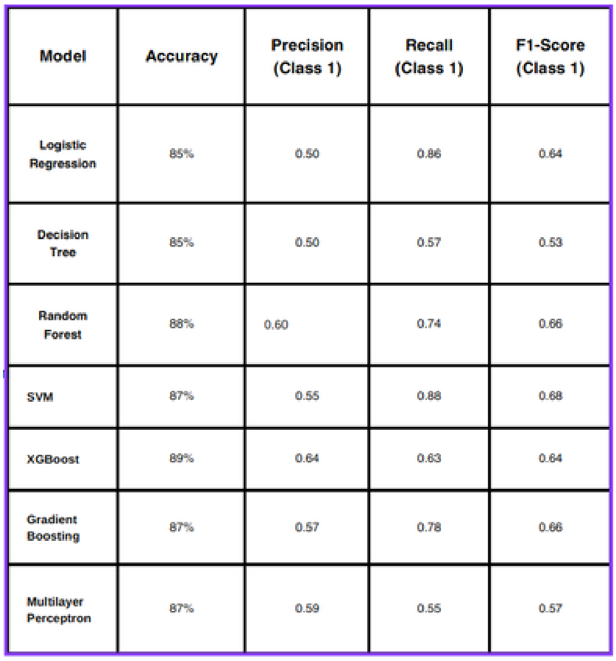
3.Loss Function:

For binary classification, MLP minimizes the binary cross-entropy loss:

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)\right]$$

6.3. Model Evaluation

We evaluated all models using accuracy, precision, recall, and F1-score. Given the imbalanced nature of the dataset, recall for the minority class (not finished) was prioritized.



Model	Accuracy	Precision (Class 1)	Recall (Class 1)	F1-Score (Class 1)
Logistic Regression	85%	0.50	0.86	0.64
Decision Tree	85%	0.50	0.57	0.53
Random Forest	88%	0.60	0.74	0.66
SVM	87%	0.55	0.88	0.68
XGBoost	89%	0.64	0.63	0.64
Gradient Boosting	87%	0.57	0.78	0.66
Multilayer Perceptron	87%	0.59	0.55	0.57

6.3 Hyperparameter Tuning:

Hyperparameter optimization was performed to enhance model performance. Two main models were tuned: Random Forest and XGBoost.

- The optimization process significantly enhanced the models' ability to identify at-risk students. For the Random Forest model, parameters were fine-tuned, resulting in a recall of 80% for the minority class. Similarly, for XGBoost, parameters like were adjusted to balance precision and recall effectively.
- While both models demonstrated strong overall performance, XGBoost outperformed Random Forest in terms of recall for the minority class (89% vs. 80%), making it a more reliable choice for identifying at-risk students. This was achieved thanks to XGBoost's ability to handle imbalanced datasets and its regularization techniques, which reduced overfitting.

6.4 Model Robustness and Generalization

To ensure that our selected model, XGBoost, generalizes well to unseen data, we conducted the following analyses:

1. **Cross-Validation Performance:**
   - XGBoost achieved an average accuracy of 90.5% across 5-fold cross-validation, indicating consistent performance across different data splits.

2.Evaluation on Noisy Data:

- We tested the robustness of the model by introducing small random noise to the test set. Despite the perturbations,it achieved an accuracy of 81%, demonstrating resilience to minor variations in input data.

Receiver Operating Characteristic (ROC) Curve

The ROC curve evaluates the trade-off between the true positive rate (TPR) and the false positive rate (FPR) across various threshold values for the predictive model. For the final model, the area under the curve (AUC) was 0.94, which indicates a strong ability to distinguish between the two classes.

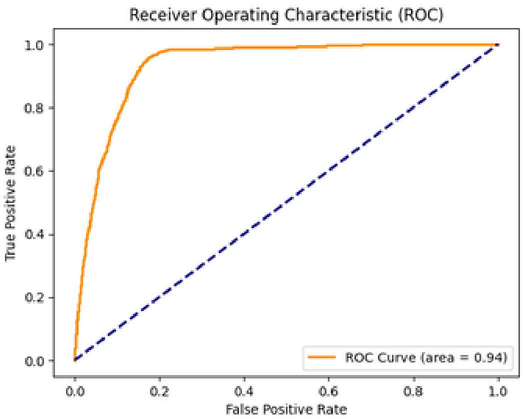


FIGURE 3. Receiver Characteristic (ROC)

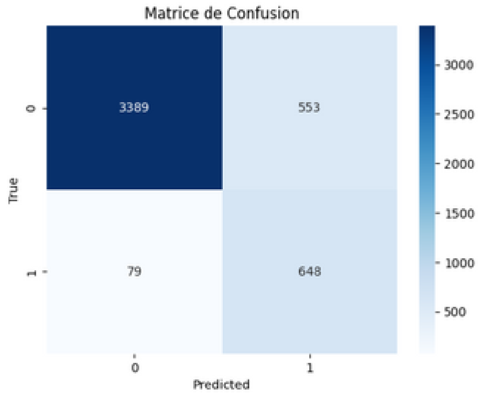


FIGURE 4.XGBoost confusion matrix

3.Comparison of Train vs. Test Performance:

- No significant performance drop was observed between the training and test sets, further supporting the generalizability of the model.

4.**Performance Metrics for the At-Risk Class:**

- XGBoost achieved a Recall of 89% for the minority "at-risk" class, which is critical for early intervention and aligns with the project's objectives.

7 .Recommender System

To provide targeted support for students identified as at risk of dropping out, a recommender system was developed. This system integrates data-driven insights with personalized recommendations to address specific challenges faced by students.

7.1 Categorization

Students were categorized into risk segments based on their performance and engagement metrics:

- **Low Engagement**: Students with minimal interaction and participation in course activities.
- **Low Performance**: Students exhibiting poor academic results or low cumulative scores.
- **High Difficulty:** Students struggling with course content, often evidenced by repeated attempts or high difficulty scores.
-

7.2 Tailored Recommendations

Each category was mapped to a set of tailored recommendations designed to mitigate their specific challenges.

7.3 Personalized Recommendations

A layer of personalized advice was added to further refine the recommendations based on additional individual factors:

- Engagement Consistency: For students with inconsistent engagement, suggestions include leveraging interactive content and setting study reminders.
- Disability Support: Students with disabilities are directed toward accessibility tools, extended deadlines, and tailored support.
- Difficulty and Learning Pace: Students facing high difficulty and slow learning pace are advised to manage workload and attend time management workshops.

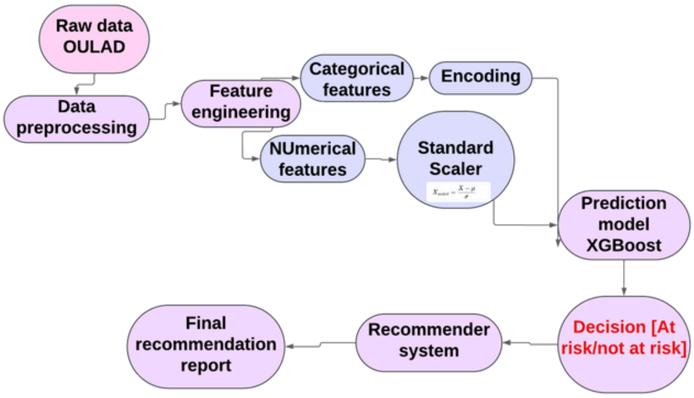


FIGURE 5. System Architecture and Workflow Overview

8.Contribution

Recommender systems and predictive models have become essential in various domains, especially in online education, to address course dropout rates. Several prior works have focused on this challenge:

1. **MOOC Dropout Prediction:**
   - Kloft et al. (2014) [1] emphasized the use of behavioral data, such as clickstream activities, to predict dropout risks in MOOCs. However, their model lacked a temporal focus to capture student engagement over time.
2. **Recommender Systems for Personalized Learning:**
   - Mustafa et al. (2020) [2] proposed collaborative filtering-based recommendation systems to personalize educational resources. While effective for general recommendations, their approach did not specifically target at-risk students.
3. **Combined Approaches:**
   - Xing et al. (2016) [3] integrated predictive models with recommender systems for real-time interventions. However, their work primarily focused on engagement and did not address other critical factors like performance and difficulty.

Our Contribution:

- Integrates multidimensional metrics, such as engagement, performance, and difficulty.
- Adapts recommendations specifically for at-risk students based on their segment.
- Combines prediction and recommendation in a unified framework tailored for early intervention.

9. Limitations

Identified Challenges

1. **Data Dependence:**
   - The model relies heavily on the structure of the OULAD dataset, limiting its application to other platforms with different data formats.
2. **Cold Start Problem:**
   - Students with insufficient historical data are harder to categorize and recommend actions for.
3. **Generic Recommendations:**
   - Although tailored, some recommendations may not fully align with individual student needs due to lack of contextual information.

10. Conclusion and Future Work.

This project addressed the challenge of high dropout rates in online learning platforms by developing a predictive model and a personalized recommender system using the OULAD dataset. The XGBoost model achieved an 89% recall for identifying at-risk students, and the recommender system provided tailored interventions based on engagement, performance, and difficulty metrics. While the results highlight the potential of these approaches, further steps are required to implement them in real-world settings.

Future Work

The system was not deployed due to the limited scope of this study, which focused on research and development rather than implementation. Future work should include deploying the system in a live online learning environment to evaluate its real-world performance and scalability. Another direction could involve adapting the model to diverse datasets to ensure generalizability. Addressing limitations, such as the cold start problem, and enhancing the recommender system with real-time feedback and contextual data, would also improve its effectiveness.

References

1. Kloft et al., "Predicting MOOC Dropout Over Weeks Using Machine Learning Methods."
2. Mustafa et al., "Personalized Learning Recommendations in MOOCs."
3. Xing et al., "Dropout Prediction and Intervention in MOOCs."