Scope
Volume 14 Number 04 December 2024

A Novel Predictive Model for University Dropout Prevention Using Machine Learning

[1]Pranab Gharai; [2]Avijit Kumar Chaudhuri; [3]Daizy Deb; [4]Arnab Chakraborty
[1]Research Scholar, 2Associate Professor, [3]Assistant Professor, [4]M.Tech. – Student
[1,2,3,4] Computer Science & Engineering, Brainware University Barasat, Kolkata, West Bengal, India

Abstract

In the higher education (HE) sector, balancing the number of students enrolled and those who pass out is a big challenge. This issue leads to the loss of potential talent and negatively affects higher education institutes in terms of finance and academics. Student dropout is a multi-factor related problem that needs a contemporary approach to identify the main factors for predicting the student who has the chance to drop out. Nowadays, applying various Machine Learning (ML) techniques in the Education sector has gained much attention from educators and education administrators. The principal research objective of the paper is to develop a machine learning approach to predict the possibility of academic failure of a student in the higher education path. The authors define academic failure as a dropout in the middle of a course and academic success as completing the course within a particular duration. So, from the ML's point of view, the study deals with classification problems, specifically binary classification. Through this study, the authors try to find the ML models for the issue by comparing the performance of different state-of-the-art algorithms based on the Enrolled Students (ES) dataset. The authors use a stacking model that uses a multilayer perceptron as a meta-classifier, Random forest and Gradient Boosting as Base Classifier, which gave better results than classical algorithms like Logistic Regression (LR), Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), Multilayer Perceptron (MLP). The developed stacking model gave us the best accuracy 87%. This model also obtains good scores for other performance metrics like sensitivity, specificity, ROC-AUC and Kappa Statistics.

Keywords: Higher Education, Dropout, Machine Learning, Logistic Regression, Naïve Bayes, Random Forest, Gradient Boosting, Multilayer Perceptron, Stacking Classifier

Introduction

In the Higher education (HE) sector, it is a big challenge to balance the number of students enrolled and several students who pass out. School scores, gender, and faculty factors are the leading causes of students dropping out of HE(Paura&Arhipova, 2014). This issue leads to the loss of potential talent and negatively affects higher educational institutes in terms of finance and academics. Student dropout is a multi-factor related problem that needs a contemporary approach to identify the main factors for predicting the student who has the chance to drop out. An individual's academic performance highly depends on their intellectual aspirations. However, the requirement or objective is driven by the education system's effectiveness and their surroundings' social and economic implications. Nowadays, applying various ML techniques in the Education sector has gained much attention from educators and education administrators. Based on student data like academic results, behaviour, demographic information, family's educational background, economic condition, etc. ML models can identify at-risk students. The early prediction helps to provide counselling and other academic support to targeted students, reducing dropout and increasing academic success.

The central research objective of the paper is to develop a machine learning approach to predict the possibility of academic failure of a student in the higher education path. The authors define academic failure as a dropout in the middle of a course and academic success as completing the course within a particular duration. So, from the ML's point of view, the study deals with classification problems, specifically binary classification. Through this study, the authors try to find the ML models for the issue by comparing the performance of different ML algorithms based on the Predict Student's Dropout (PSD) dataset.

The authors divide the whole experiment process into two phases. In the initial phase, the authors apply different contemporary classification models on the PSD dataset and record the model accuracy and other performance criteria of classifiers. The authors use Logistic Regression (LR), Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), and Support Vector Machine(SVM) classifiers in the initial phase. The authors build a StackedModel (SM) in the second phase. The authors have also made a comparative review between classical models and the proposed model. Figure 2 shows the flowchart of step-wise approach to constructing the model.

In the first step, the dataset has been divided into two subsets – a training set and a testing set using train-test split techniques. The training dataset is fed to the ML model for training the model. The testing dataset is used to evaluate the performance of an ML model for unknown data. Regarding train–test split techniques, standard ratios - 50% - 50%, 66% - 34%, and 80% - 20% were used. The authors also applied 10-fold cross-validation to test the general applicability of the model.

The second step is the model training and testing step some state-of-the-art ML models were used and train the models through the training dataset. During this step, the models learn to understand the patterns and relations in the training dataset. Two types of models are mainly selected for training purposes. Firstly, the state-of-the-art classifiers were used, where models predict the output based on a single ML algorithm such as LR, NB, SVM, and DT. Secondly the ensemble models like RF and proposed SM.

After training and testing the state-of-the-art ML models, analyze the model's performance in the third step. In the study, the authors use performance matrices such as confusion matrix, accuracy, sensitivity, specificity, receiver operating characteristic (ROC), the area under the curve (AUC), and a statistical measure, Kappa statistics. These metrics provide different angles of the model's performance.

Moreover, in this research work on student dropout, the authors purposely avoid choosing particular features to focus on because some important features may be excluded if done so. Therefore, the study will retain all the available features to provide a complete picture of student dropout factors, not leaving out any critical data. This approach enables the identification of several variables in dropout, which could not have been easily seen or considered in prior studies due to the issue's complexity with influences from so many factors. Therefore, the authors keep the attribute space as rich as possible to capture potentially interesting patterns and dependencies inherent in the features.

Relevant literature

There are numerous causes of stress for which several students experience anxiety during their college or university life, such as pressure from academic work, financial pressures, and loneliness. Studies show that "at risk learners" who prematurely exit higher education due to these challenges end up in anti-social activities or low-wage employment, which creates cycles of social/ economic marginalization (Bowman et al., 2024; Olmedo-Cifuentes & Martínez-León, 2022). These outcomes not only do not prepare them for integration into society but also contribute to other social problems; hence, interventions should be directed to promote and enhance student retention.

In this section, we will provide overviews of these areas, which have been reviewed and investigated by different researchers and relate to the use of machine learning approaches to tackle student dropout in HEs. These studies use various complex prediction algorithms and data analysis aimed at finding predictors and risks of students' dropouts and containing them. According to multiple strategies based on different datasets, including students' academic results, attendance data, demographics, and behavioural data, machine learning models appear to accurately predict the risks of dropping out. This new area of study expands the knowledge base of students' withdrawal from education and offers practical recommendations for policy decisions and teaching practice regarding strategies for decreasing dropout rates in higher education.

Prenkaj et al., 2020 show in their study that ML is a promising approach for predicting student dropout. Some ML models such as DT, RF, SVM, and neural networks efficiently identify essential factors of student dropout issues and also mark out students at risk of dropout at an early stage.

In their study, Segura & Hernández, 2022 used various approaches and methods. First, they followed a feature selection process to discover which features correlated more with the dropout rates. They then used an SVM, DT, Artificial Neural Network (ANN), and LR. The results indicate that the models do not rely on enrolment characteristics for dropout prediction only and obtain a higher accuracy when the first-semester performance of the student is incorporated. However, although student performance continues to be significant, there can be other parameters, including a student's level of affinity for the course the student chose. Several of these practices differ in efficiency depending on the program's areas of emphasis. While the best-performing models are those of the ML family, a simple LR can be used as a baseline.

Kemper et al. (2020) propose two machine learning models, logistic regression and decision trees, to identify early dropout students in KIT universities. The developed models are based on data derived from examination results, which, in most universities, do not usually present a problem regarding the availability of data that needs to be collected. For this reason, the proposed method is viable and can be applied to other institutions easily. This study establishes that DTis more accurate than LR models in predicting the values by a slight margin. Nevertheless, both methods yield high prediction accuracies, increasing to 95% after three semesters of prediction. Surprisingly, this enables the classification accuracy to be greater than 83% as early as the first semester.

The authors incorporated additional literature on the ML classifiers used in this study in the relevant sections.

SlNo	Papertitle	KeyFinding	Methodology	Accuracy(%)
1	Beaulac & Rosenthal, 2019	1. PredictingProgram Completion 2. PredictingtheMajor	RF	91.19 47.41
2	Wan Yaacob et al., 2020	The aim of this study is to identify the key determinants of undergraduate student dropout rates in Computer Science program of University Technology MARA and to determine which data mining technique is more suitable to find these key determinants.	LR	90.8

3	Ndou et al., 2020	Educational data mining can be used to predict student performance. A variety of factors can influence student performance, including socioeconomic factors, psycho-social factors, pre- college and intra-college scores, and individual attributes.	NB, DT, SVM, ANN	84.60
4	Martins et al., 2021	To build ML classification models to predict student thatmightbeatriskoffailin g to succeed in finishing them degreesindue time.	RF Extremegradient boosting	72.0 73
5	Panagiotakopoulos et al., 2021	How can supervised machine learning algorithms be used to effectively predict student dropout in MOOCs (Massive Open Online Courses) at an early stage?	LightGBM Stacking	95 96
6	Yağcı, 2022	The first parameter was the prediction of academic performance based on previous achievement grades. The second one was the comparison of performance indicators of machine learning algorithms.	RF	84.9
7	Patricio Rodríguez et al., 2023	Develop and evaluate a ML model to effectively predict student dropout in school systems.	Decision tree ensembles (XGBoost)	93

Table 1: Comparative study of relevant literature

Methodology

The main research question of this paper is to develop an appropriate machine learning model following the prevention of academic failure in students of higher learning institutions. The authors define dropping out mid-course as failure and completing the course within a fixed period as a success. From a Machine Learning perspective, the study seeks to solve a classification problem of the binary type. The authors want to suggest which ML models might solve this problem by using the PSD dataset to test the state-of-the-art algorithms' attempts. These are all presented in the experimental workflow depicted in Figure 1 below.

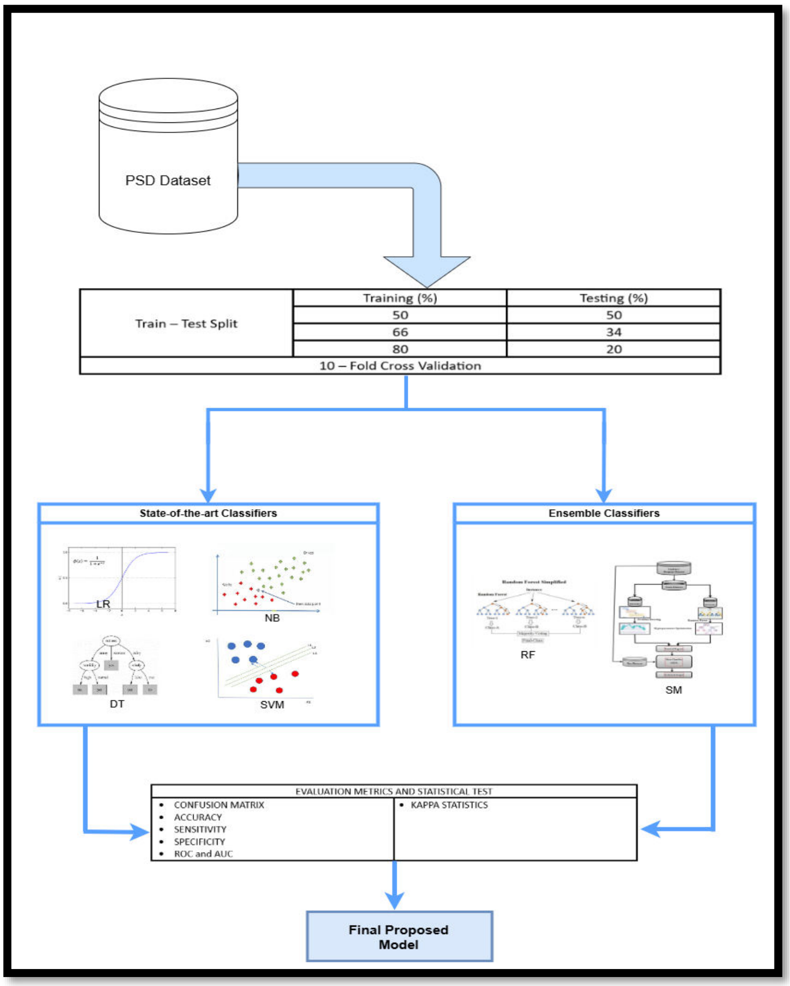

Figure 1: Experimental workflow

This research delves into five popular ML methods, from RT to SVMs. We dissect their strengths, weaknesses, and performance across diverse situations. Our goal is to unlock their full potential and make them work better, in more ways, for everyone. It's a journey towards smarter, more powerful AI.

Logistic regression

LR is one of the most popular techniques for analyzing binary classificationThus; it helps address student dropout issues. The results of LR can include students who performed poorly, were frequently absent, were from low socio-economic backgrounds, and were unmotivated, so the students who drop out can be predicted. Tinto, 1993 identified students' academic and social integration as potential retention predictors; more recently, García-Sánchez, 2020 used LR to determine dropout risk. These models base themselves on probabilities that involve students' scores, conduct, and other factors to enable institutions to foster necessary corrective measures (Alon & Tienda, 2005). Such modelling assists educational institutions in acting early enough, preventing student loss or poor results.

Naïve bayes

The NB algorithm has been used frequently to predict the probability of student dropout because of its efficacy in classification problems. The model operates based on Bayes' theorem, which requires naïve independence between the features. While used in student dropout prediction, NB can predict a student's ability to drop out or continue studies based on performance, attendance patterns, socioeconomic status and activity levels. Research has revealed that NB outperforms most rudimentary ML algorithms in the area of specific student dropout prediction while being almost as effective as complex algorithms (García et al., 2020; Sarker et al., 2021). Furthermore, it is most convenient when working with data that has categorical variables and when a model that allows making fast and precise predictions in real time is required (Chaurasia & Pal, 2018).

Decision tree

Student dropout rates in institutional environments have been analyzed and predicted using the DT approach because of the model's capacity to handle non-linear interactions between factors. For student retention, DT can define and locate significant factors contributing to student dropouts, such as academic achievement, socioeconomic status, truancy, and participation in school activities (Tharwat, 2016). These models assist educators and administrators in creating appropriate intervention strategies since they outline the decision rules for the risk of dropout based on past records (Tena & Henson, 2016). The most essential benefits of DT include interpretability and applicability to both continuous and nominal data, making decision trees suitable for early intervention strategies in students at risk samples (Liu & Wu, 2017). In addition, several ensemble techniques may generalize the basic forms of DT, such as using RF to enhance predictive performance (Breiman, 2001).

Random forest

RF, an ensemble learning technique based on DT, holds a potent promise in tackling the problem of student dropout rates as large datasets can be effectively handled and predictive accuracy can be enhanced. RF is robust in handling nonlinear data and interactions between variables, including academic performance, attendance, socio-economic status, and level of extracurricular participation presented in the model by constructing several DT consecutively and using their results (Liu et al., 2019). It has been applied in several educational settings to determine which students are more prone to developing undesirable behaviours and to detect the appropriate time to prevent those students (Zhou et al., 2020). RF is less prone to overfitting, which liberates the model's full potential when data contain noise or are not complete (Breiman, 2001). Also, RF gives valuable information on feature importance; thus, it assists in determining which factors affect positively or negatively student retention and dropout (García et al., 2018). The capability of using the software for both categorical and numerical forms of data likewise adds to its versatility in real-life educational environment applications.

Multilayer perceptron

MLP is a type of artificial neural network that is well suited for predicting student dropout as it can account for the complexity of the non-linear relationship between the input characteristics. MLPs are composed of several levels of neural networks where the network can identify complex data features like performance grades, student conduct, and student demographics to predict dropout risk factors (Alam et al., 2020). These

networks are essential in educational data mining because they do not require researchers to manually identify relations and structures, which might be invisible to traditional statistical analysis by He & Xu (2019). Concerning student retention, MLPs have shown increased accuracy in predictions better than DT or LR while handling considerable datasets with high numbers of inputs (Kaur & Singh, 2021). However, MLPs may suffer from overfitting problems, although this significant shortcoming is correctable by adequate hyperparameters adjustment and modern tools for avoiding overfitting, such as dropping out layers of neurons (Srivastava et al., 2014). Nevertheless, MLPs remain valuable for early students' identification and timely retention intervention.

Stacking model

In this research, the stacking model is at the stacking level of the architecture, which performs better by combining multiple classifiers in a structured manner. Below is a detailed explanation of the approach:

Data Preparation: The proposed research employs the train-test split on preprocessing, utilizing 10-fold cross-validation for enhanced model validation. Before the actual feature rescaling process, a StandardScaler is used to rescale the feature values to range between 0 and 1 because of the input requirement necessary for many of the models in this study.

Base Learners: Two basic classifiers are chosen—RF and GB. These models are tuned using Grid Search CV to determine the best hyperparameters by cross-search from a set of values.

Stacking Classifier: Finally, the optimized output of the base models is fed to a Stacking Classifier for the final predictions. During the training phase, each base model forecasts the training data, and the results are used to form a new training set. Subsequent stages of the stacking process are based on this dataset.

Meta-Learner: The last estimator in this model is the MLP Classifier, which operates on a dataset created by the outputs of the base models and tries to learn the optimal way to combine such predictions. The meta-learner decides how the base models should be used to achieve the least classification error.

Prediction and Evaluation: Finally, an out-sample assessment evaluates the fitted stacking model on a new test data set. The model performance is evaluated using accuracy, confusion matrix, ROC AUC and Cohen's Kappa.
This hierarchical system of operating with numerous learning algorithms adds up to classification accuracy, so stacking is highly effective in machine learning. Figures of the experimental design, flowcharts and model building are provided in Figure 1 and Figure 2, respectively.

Algorithm
Stacked Ensemble Model with Pre-processing and Cross-validation

Step 1. Preprocessing of the Dataset

- **Train-Test Split**:
  - $The\ dataset\ D = \{X, y\} is\ first\ divided\ into\ a\ training\ set\ D_{train} = \{X_{train}, y_{train}\}\ and\ a\ test\ set\ D_{test} = \{X_{test}, y_{test}\}$
  - $The\ training\ set\ is\ partitioned\ into\ thirty\ subsets, denoted\ as\ D_{train}^{(1)}, D_{train}^{(2)}. \ .\ .\ .\ , D_{train}^{(30)}$
  - $For\ each\ fold\ i, the\ model\ is\ trained\ on\ D_{train}^{(-i)} = D_{train} \backslash D_{train}^{(i)}\ and\ validated\ on\ D_{train}^{(i)}$
- **Feature Scaling**:
  - $Standard\ Scaler\ is\ applied\ to\ the\ features\ X\ to\ ensure\ that\ each\ feature\ is\ centred$

- $(mean = 0) and scaled (variance = 1): X_{scaled} = \frac{X-\mu x}{X - \sigma x} represent the mean and standard deviation of X, respectively.$
- $This\ operation\ ensures\ all\ features\ are\ on\ the\ same\ scale, which\ is\ important\ for\ many\ models, including\ neural\ networks\ and\ distance - based\ algorithms.$

Step 2. Base Classifiers

- **Random Forest Classifier**:
  - Random Forest (RF) is an ensemble of decision trees. The final random forest prediction is obtained by aggregating the predictions of individual trees (via majority voting for classification problems).
  - $The\ mathematical\ formulation\ for\ a\ single\ tree's\ prediction\ is\ f_{tree}(x) = sign(\sum_{i=1}^{N} w_i \,.\, h_i(x)) where\ h_i(x) is\ the\ prediction\ of\ tree\ i, and\ w_i\ is\ the\ weight\ of\ the\ i - th\ tree$
- **Gradient Boosting Classifier**:
  - $Gradient\ Boosting\ builds\ a\ sequence\ of\ weak\ learners\ (typically\ decision\ trees)\ where\ each\ subsequent\ tree\ corrects\ the\ errors\ made\ by\ the\ previous\ one.$

$$The\ model\ can\ be\ formulated\ as\ F(x) = \sum_{m=1}^{M} \gamma_m h_m(x)\ where\ h_m(x)\ is\ the\ prediction\ of\ the\ m - the\ weak\ learner\ and\ \gamma_m\ is\ the\ corresponding\ weight.$$

Step 3.Hyperparameter Tuning with GridSearchCV

- **GridSearchCV** is used to search over a specified parameter grid to find the best combination of parameters for each base model:
  - Define a grid of hyperparameters $P = \{p_1, p_2, \ldots., p_k\}$ for each classifier.
  - Perform cross-validation on each combination of hyperparameters p_i to identify the optimal configuration:

$$\hat{P} = \arg\min_{p_i} \text{CV Loss}(p_i)$$

Where $\text{CV Loss}(p_i)$ is the average cross-validation error for a particular setting of hyperparameters.

Step 4. Stacking Classifier

- **Training the Base Learners**:
  - Once the base classifiers (Random Forest and Gradient Boosting) are tuned, they are trained on the full training data X_{train}. Each base classifier produces predictions $\hat{y}_{RF}(X)$ and $\hat{y}_{GB}(X)$

- These predictions form a new dataset $X_{stack} = [\hat{y}_{RF}(X), \hat{y}_{GB}(X)]$
- **Meta-Learning (Final Estimator)**:
    - A **Meta-Learner** (Multi-Layer Perceptron Classifier-MLPClassifier) is trained on the new dataset X_{stack}.
    - The goal of the meta-learner is to learn the best combination of the base classifiers' predictions to minimize the total classification error.
    - The output of the final meta-learner is:

$$\hat{y}_{stack} = \text{MLPClassifier}\left(X_{stack}\right)$$

The described approach combines multiple machine-learning techniques into a cohesive pipeline to optimize classification performance. The working of the proposed method is depicted in Figure 2.

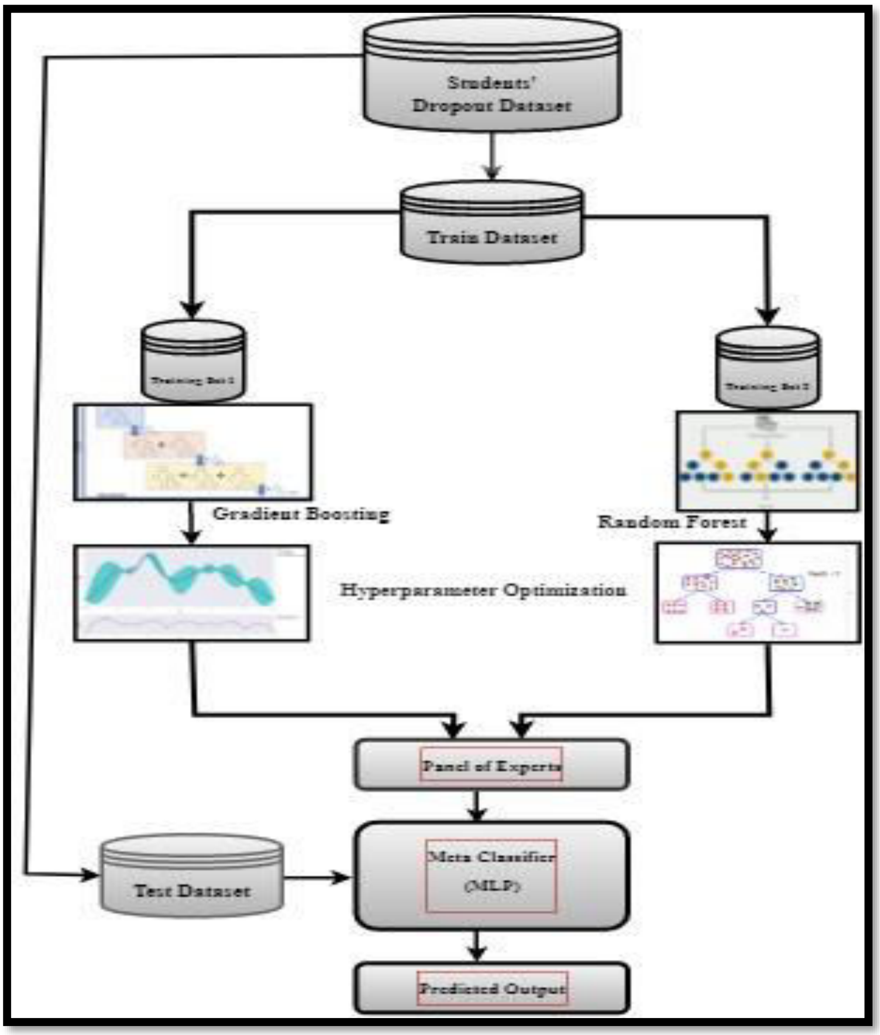

Figure 2: Proposed Stacking Model

Performancemetrics

<table>
<tr><th>S/N</th><th>Metrics</th><th>Formula/Description</th></tr>
<tr><td>1</td><td>Confusion Matrix</td><td>
<table>
<tr><td></td><td></td><td colspan="2">Actual</td></tr>
<tr><td></td><td></td><td>Without CVD</td><td>With CVD</td></tr>
<tr><td rowspan="2">Predicted</td><td>Without CVD</td><td>True Positive (TP)</td><td>False Positive (FP)</td></tr>
<tr><td>With CVD</td><td>False Negative (FN)</td><td>True Negative (TN)</td></tr>
</table>
</td></tr>
<tr><td>2</td><td>Accuracy</td><td>$\frac{TP+TN}{TP+TN+FP+FN}*100$</td></tr>
<tr><td>3</td><td>Sensitivity</td><td>$\frac{TP}{TP+FN}*100$</td></tr>
<tr><td>4</td><td>Specificity</td><td>$\frac{TN}{TN+FP}*100$</td></tr>
<tr><td>5</td><td>Kappa statistics</td><td>$\frac{pa-pac}{(1-pac)}$,‘pa’ represents total agreement probability and ‘pac’ represents probability ‘by chance’. Its range is (-1,1).</td></tr>
<tr><td>6</td><td>Area under the curve (AUC)</td><td>Receiver operating characteristic (ROC) is plotted between Sensitivity and (1-Specificity). The area under the curve (AUC) measures the degree to which the curve is up in the northwest corner.</td></tr>
</table>

As shown in Table 3, Table 4, Table 5, Table 6, Table 7,Table 8, and Table 9 the machine-learning approaches yielded different levels of accuracies, precision, sensitivity,specificity, f score, kappa statistics,and AUC-ROCrespectively.

To assess the performance of the proposed work, we employ a tool called a Confusion Matrix. This chart helps us determine how well a machine-learning model is performing by tracking four key metrics:

True Positive (TP): The number of students with dropouts is classified accurately as dropouts.

False Positive (FP): To be suited for a diagnosis of dropouts, a student would have to be among those without such matters who are wrongly identified as suitable for it.

True Negative (TN): The number of subjects classified adequately as not having dropout problems.

False Negative (FN): Those who do not have dropouts are diagnosed with not having them.

These metrics enable us to ascertain how effectively we can identify students with dropouts relative to others.
Key performance measures derived from these terms include:
Accuracy: The percentage of all the correct outcomes; the sum of the above-mentioned TP and TN divided by the total number of predictions. This is like knowing how many times the model got it right overall.
Precision: This measure how proper the model is in predicting someone having a dropout. It informs us how many students classified as dropouts have those issues.
Recall (or Sensitivity): This shows the ability of the model to correctly capture all the individuals with dropouts. It also informs us about the extent to which participants presenting with dropout concerns were accurately diagnosed.

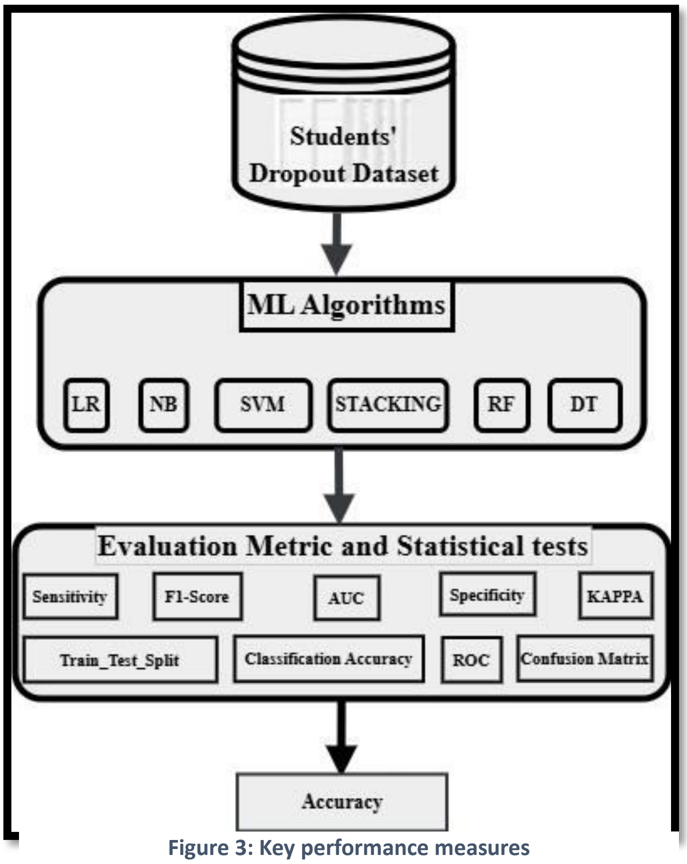

Figure 3: Key performance measures

Specificity: This gives a measure of accuracy by which the model can correctly classify those who do not have dropouts. It shows us the likelihood of negative cases in all persons without dropouts.

F1 Score: This is a trade-off between precision and recall. It aids in finding that 'middle ground' between correctly identifying dropouts and avoiding exclusions of anyone significant. This is determined using a unique method with accuracy and completeness as the measures used in its computation.

AUC-ROC Curve: The AUC-ROC curve is a metric used to assess the model Classifier / Probability regarding two outcomes, such as dropout or not. They evaluate the model's validity regarding how well it can differentiate between these outcomes. The "ROC" part is concentrated on the search for an adequate value of correctly classified dropout cases (True Positives) and the minimum rate of false alarms (False Positives). The AUC value is from 0 to 1 and measures the performance similar to the score – the higher the value, the better the model at differentiating cases.

Kappa Statistics: As implicit in its name, Kappa statistics is a statistic designed to measure the degree of credibility of two or more raters or judges. It aids in establishing if their agreement is more than random, thus explaining how well they are in agreement independently.

Dataset description

A dataset used in this study is collected from the UCI Machine Learning Repository. A higher education institution collected the dataset from many separate databases about students enrolled in various undergraduate programs, including agronomy, design, education, nursing, journalism, management, social services, and technology. The dataset contains data on socioeconomic characteristics, academic routes, demographics as of enrolment, and the student's final academic standing after the first and second semesters. Classification models are constructed using the data to forecast students' academic success and dropout rates. The dataset consists of 4424 instances, with 36 nominal independent attributes. The only dependent attribute, "Target," contains two class values, "Dropout" and "Graduate." The following table presents a description of all the data fields.

No.	Features	Total data	Missing data	Data type
1	Marital status	4424 (100%)	0 (0%)	Nominal
2	Application mode	4424 (100%)	0 (0%)	Nominal
3	Application order	4424 (100%)	0 (0%)	Nominal
4	Course	4424 (100%)	0 (0%)	Nominal
5	Daytime/evening attendance	4424 (100%)	0 (0%)	Nominal
6	Previous qualification	4424 (100%)	0 (0%)	Nominal
7	Previous qualification grade	4424 (100%)	0 (0%)	Nominal
8	Nationality	4424 (100%)	0 (0%)	Nominal
9	Mother qualification	4424 (100%)	0 (0%)	Nominal
10	Father qualification	4424 (100%)	0 (0%)	Nominal
11	Mother occupation	4424 (100%)	0 (0%)	Nominal
12	Father occupation	4424 (100%)	0 (0%)	Nominal

13	Admission grade	4424 (100%)	0 (0%)	Numerical
14	Displaced	4424 (100%)	0 (0%)	Nominal
15	Educational special needs	4424 (100%)	0 (0%)	Nominal
16	Debtor	4424 (100%)	0 (0%)	Nominal
17	Tuition fees up to date	4424 (100%)	0 (0%)	Nominal
18	Gender	4424 (100%)	0 (0%)	Nominal
19	Scholarship holder	4424 (100%)	0 (0%)	Nominal
20	Age at enrolment	4424 (100%)	0 (0%)	Numerical
21	International	4424 (100%)	0 (0%)	Nominal
22	Curricular units 1st sem credited	4424 (100%)	0 (0%)	Numerical
23	Curricular units 1st sem enrolled	4424 (100%)	0 (0%)	Numerical
24	Curricular units 1st sem evaluations	4424 (100%)	0 (0%)	Numerical
25	Curricular units 1st sem approved	4424 (100%)	0 (0%)	Numerical
26	Curricular units 1st sem grade	4424 (100%)	0 (0%)	Numerical
27	Curricular units 1st sem without evaluations	4424 (100%)	0 (0%)	Numerical
28	Curricular units 2nd sem credited	4424 (100%)	0 (0%)	Numerical
29	Curricular units 2nd sem enrolled	4424 (100%)	0 (0%)	Numerical
30	Curricular units 2nd sem evaluations	4424 (100%)	0 (0%)	Numerical
31	Curricular units 2nd sem approved	4424 (100%)	0 (0%)	Numerical
32	Curricular units 2nd sem grade	4424 (100%)	0 (0%)	Numerical
33	Curricular units 2nd sem without evaluations	4424 (100%)	0 (0%)	Numerical
34	Unemployment rate	4424 (100%)	0 (0%)	Numerical
35	Inflation rate	4424 (100%)	0 (0%)	Numerical
36	GDP	4424 (100%)	0 (0%)	Numerical
37	Target	4424 (100%)	0 (0%)	Numerical

Table 2: Dataset description

Results and Discussion

The authors tried to propose a stacking model in this research paper to identify the risk of dropping out of a student. Dropping out of college means leaving the course without completing it. This paper tried to approach the problem through machine learning techniques. From anML's perspective, this paper focuses on the binary classification, predicting one of the two classes (either success or failure). The authors compared many machine learning algorithms, even though they developed their stacking algorithm that gave good results as compared to others. They tested classifiers like DT, NB, LR, SVM and RF with their proposed stacking model that consists of MLP as meta classifier and RF and Gradient Boosting as base classifier.

The experiment showed that the stacking model gave the best results as compared to other classical algorithms. This model also performed well in ROC-AUC (a measure of the model's ability to distinguish between classes), specificity and sensitivity.

Overall, with the help of a ML model, authors tried to predict which students are likely to drop out of higher education. By predicting it educators and administrators can intervene in time to stop the early dropout of the students from college. It is going to increase the success rate.

Accuracy						
	LR	NB	SVM	DT	RF	SM
10 Fold	82	82	67	80	87	87
50 - 50	81	83	86	80	86	87
66 - 34	82	83	85	80	86	86
80 - 20	80	82	85	78	86	86

Table 3: Accuracy of different ML models

The accuracy results of this study show that the RF and SM consistently outperformed the other classifiers across all data splits, achieving the highest accuracy of 87% in the 10-fold, 50-50 split, and in 66-34 and 80-20 splits it is 86%. The proposed SM achieved the highest accuracy of 87% in the 10-fold, 50-50, and 86% in the 80-20 and 66-34 split. LR and NB demonstrated stable performance, with accuracy ranging from 80% to 83%, while SVM exhibited lower performance, especially in the 10-fold cross-validation with an accuracy of just 67%. DT also performed moderately well, though its accuracy ranged from 78% to 80%, and was more sensitive to smaller training sets. The Stacked Model, which integrates multiple classifiers, consistently achieved superior performance, highlighting the benefits of ensemble methods for improving prediction accuracy. Overall, ensemble models like RF and SM showed robustness across varying data splits, making them preferable for classification tasks in this study.

Precision						
	LR	NB	SVM	DT	RF	SM
10 Fold	83	82	46	81	87	87
50 - 50	81	83	86	80	86	87
66 - 34	82	83	85	80	85	86
80 - 20	81	82	85	78	86	86

Table 4: Precision score of ML models

As shown in the precision table of the study, the SM achieves better accuracy than the individual classifiers in all the partitions. In particular, the SM yielded the highest results for the 10-fold, 50-50; overall accuracy was equally as high – 87%, while the 80-20 split brought a slightly lower accuracy, 86%. Of all the individual classifiersRF had the best result, scoring 87% in the 10-fold split and 86% in the 50-50 split by SVM. On the other hand, LR, NB, DT, and SVM performed comparatively lower in general, having an accuracy of trains varying between 46% to 86% for different splits of datasets.

From these outcomes, it is clear that state-of-the-art classifiers like RF, SVM, and others are present, but the SM is better because it incorporates more than one classifier whose potential is captured by the SM than any individual classifier.

Recall / Sensitivity						
	LR	NB	SVM	DT	RF	SM
10 Fold	82	82	67	80	87	87
50 - 50	81	83	86	80	86	87
66 - 34	82	83	85	80	86	86
80 - 20	80	82	85	78	86	86

Table 5: Sensitivity of different ML models

These recall (sensitivity) results show that the SM performs best in most data splits, with 87% and 86% recall for 10-fold and 50-50, 66-34 and 80-20, respectively. Details of Table 5 showed a relatively high recall of true positives; RF had the highest recall rate of 87% in 10-fold cross-validation and 86% in other splits. SVM likewise showed a high recall rate of 67 to 86 %, whereas recall for different methods such as LR, NB, and DT was generally low, with the best results ranging from 80 to 83%. These findings also support work highlighting that while individual classifiers such as RF and SVM can yield good performances, this is further improved through combining classifiers in the SM, which provides greater versatility and accurately identifies positive instances across all data partitions. This shows how ensemble methods are better regarding recall rates than single models used in the study.

Specificity						
	LR	RF	NB	SVM	DT	SM
10 Fold	82%	87%	86%	67%	86%	88%
50 - 50	82%	83%	74%	82%	69%	84%
66 - 34	85%	83%	75%	82%	70%	82%
80 - 20	85%	88%	77%	85%	70%	85%

Table 6: Specificity of different ML models

From the above study, the researchers have not restricted the performance analysis of classification models to accurate optimistic prediction. The exact parameters used to explain how the ML models are performing to predict the true-negative instance and the specific student level that will not drop out are also important. The closer to 1, the higher the specificity, which means the classifiers are acceptable because they can correctly classify negative instances. This ability of a classifier also decreases the number of false positives, as pointed out above. For this research, the authors use the specificity metric. Based on the specificity analysis, it was found that SM has the highest specificity at 88% while the RF model has 87%, which is the second highest achieved by a 10-fold CV. The RF and the author's proposed SM also achieve a good specificity score in train-test-split. The other models all presented a satisfactory specificity score of 67-86% after using the train-test-split.

F1 Score						
	LR	NB	SVM	DT	RF	SM
10 Fold	83	82	54	80	87	87
50 - 50	80	83	86	80	86	86
66 - 34	81	83	85	80	85	86
80 - 20	79	82	85	78	85	86

Table 7: F1- score of different ML models

The evaluation using the F1 score shows the SM tends to yield the relatively highest results throughout all the data split tests, scoring an F1 of 87 and 86 % for both 10-fold and 66-34/80-20/ 50-50 split, respectively. The second best was the RF for analysis in the 10-fold cross-validation test, yielding an F1 of 87% and 85% in the other splits, showing a good balance between precision and recall. SVM had promising results, the F1 score of which varied from 54% in the 10-fold cv, 86% in the 50-50, 85% in 66-34 and 80-20 splits. The SM, as well as RF, also outperformed LR with an F1-score between 78 % and 83 % to demonstrate that these models are also viable but unable to achieve the same degree of preciseness as well as recall as the SM and RF. In general, the results confirm the effectiveness of the SM that, by incorporating multiple classifiers, achieves better F1 values, indicating the model's potential to fine-tune the recall and precision factors more successfully than single models.

Kappa Statistic						
	LR	NB	SVM	DT	RF	SM
10 Fold	56	59	0	56	69	70
50 - 50	52	60	67	53	66	68
66 - 34	55	60	65	54	66	67
80 - 20	53	60	65	52	67	68

Table 8: Kappa Statistic of ML models

The Kappa statistic results indicate the level of agreement on the classification between the predictions and the actual classification while allowing for a deal by pure chance. The SM achieved the highest Kappa all through the K-Fold splits, with 10-Fold Kappa scoring 70%; in the case of train-test partitions for 50-50, it is 68 %; for 80-20, 68%, and 66-34, it is 67%. This implies much better performance in terms of classification agreement than separate models. Another relatively accurate model was identified as RF, with the Kappa values for the accuracy falling between 66 and 69 % – therefore, there was moderate consistency between the numbers and the applied classifications. It is possible to assume some degree of acceptable variance as LR, NB, and DT have varied

Kappa scores from 52% to 60%, which translates to moderate to good inter-annotator agreement, but still worse compared to the first experiment setup where SVM had a Kappa 0% in the 10- fold-CV which means they are not in a reasonable degree of agreement To support these conclusions, the subsequent results were obtained, proving the suitability of the SM for increasing the homogeneity and overall reliability of classification decisions by using the synergy of multiple classifiers and avoiding shortcomings characteristic of individual models. The results show that using ensemble methods such as stacking significantly improves the classifier's accuracy, taking chance rates into account.

AUC Score						
	LR	NB	SVM	DT	RF	SM
10 Fold	86	85	84	78	91	92
50 - 50	73	80	83	76	82	83
66 - 34	75	80	81	77	81	82
80 - 20	74	79	81	75	82	83

Table 9: AUC score of different ML models

This overall model discrimination ability is further supported by the AUC scoresindicating that the SM yields a higher discrimination capacity than the individual classifiers. When implemented in the 10-fold cross-validation, the performance of the SM outperformed the RF at 91% AUC and LR and NB at 86% and 85%, respectively. The performance of SVM was also satisfactory; it achieved an AUC of 84%, which again was slightly worse than the SM and RF performances. The 50-50, 66-34, and 80-20 splits of the dataset resulted in maintaining the AUC of the SM at 83% in all the splits, emphasizing the model's stability across partitions. The performance of RF was also comparable with the best AUC scores between 82 – 91% but failed to beat the SM at any fold. The performance metrics of LR, NB, and DT had lower AUC scores ranging between 73-80%, which makes these models less capable of helping to differentiate between classes and the ensemble. Such findings indicate that although RF and SVM show impressive accuracy of results in their performances, the SM exhibits higher AUC coefficients in all splits and, therefore, is a better classifier of positive and negative cases and is, thus, considered the most accurate model in terms of classification.

Conclusion

Student dropout remains a significant concern in HE institutions worldwide, as it affects the institution's academic output and generates financial risks for the institution. This study aimed to establish the feasibility of using machine learning (ML) in determining the probability of student dropout to help institutions develop strategies to prevent such cases. To achieve this, the authors decided to work only with two variables: course completion in a particular period would be considered part of academic success. In contrast, dropout during a course would be regarded as academic failure. Our goal was to create an ML model that would provide practical observations that could be useful for higher education managers.

In the present work, the authors contrasted four different state-of-the-art ML algorithms, including the classical benchmarks such as LR, NB, DT, and SVM and ensemble learning methods like RF and GB. We applied a stacking ensemble model to consolidate the prediction using RF and GB as base classifiers. MLP worked as a classifier after analyzing the performance of these algorithms individually. This stacking model exceeded the individual models, yielding an accurate figure of 87% with other performance indicators, including sensitivity, specificity, ROC-AUC, and Kappa Statistics. The data analysis shows that the stacking model can accurately predict students' dropout and is more efficient than the classical model in this particular case.

The stacking model's high accuracy rates and stability state the method's efficiency and the potential of applying it in the educational field to help institutions by providing a better tool to determine students who might drop out. The analyses make it possible to forecast high risks of failure that, in turn, enable educational institutions to provide timely and specific interventions like academic counselling, tutoring, or modifying academic needs to retain students. Furthermore, the study shows the use of sophisticated ML tools to gather rich patterns in the data of the students that other, more straightforward methods may fail to detect.

This research strives to explain some of the most complex issues in the education sector and how machine learning can solve them. The model, therefore, is designed to address various aspects such as academic achievement, demographic information, Students' interactions, and behaviors, and it serves as a good platform to identify students who are likely to excel or fail. This prediction capacity could be revolutionary for administrators, as far as better student retention rates are concerned; it could be revolutionary for students, too, as far as predictors could help them solve academic difficulties more effectively.

At the same time, the work also highlights several further research directions. The stacking model applied in this investigation gave relatively high prediction accuracy. However, carefully tuning hyperparameters and extending the set of possible predictors, including socio-economic or mental health status, could improve the results. Secondly, the model's applicability in any learning environment and content area must be researched further. More complicated and advanced techniques like deep learning models or combined models could be used to improve preventive performance in the future. In addition, a significant area of concern related to practical or moral issues is the equitable use of such predictive models and the protection of student data, among other matters of ethical importance.

In conclusion, this study has demonstrated the potential of machine learning in addressing one of the most crucial issues in higher education today: student dropout. The herein-developed stacking model provides a strong predictive model of academic failure, and its effectiveness, if applied, can significantly improve retention mechanisms. Applying more advanced data analytics techniques in decisions about student engagement, higher education institutions will be right on their way to implementing a fundamentally data-driven approach to student persistence and success to increase the number of learners who complete their degrees and accomplish academic outcomes.It is evident that as machine learning advances, it can transform the learning landscape and help institutions rethink how learning support is delivered to students to support sustainable education solutions better.

References

1. Alam, M., Sadeghian, A., & Singh, P. (2020). Predicting student dropout using multilayer perceptron models. International Journal of Educational Data Mining, 12(2), 45-59.
2. Alon, S., & Tienda, M. (2005). Assessing the influence of college contexts on degree completion. Sociology of Education, 78(4), 294-317.
3. Beaulac, C., & Rosenthal, J. S. (2019). Predicting university students' academic success and major using random forests. Research in Higher Education, 60, 1048-1064.
4. Bowman, S., Morris, T. T., Dickson, M., Rice, F., Howe, L. D., & Hughes, A. M. (2024). Maternal depressive symptoms and young people's higher education participation and choice of university: Evidence from a longitudinal cohort study. Journal of Affective Disorders, 344, 339-346.
5. Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5-32.
6. Chaurasia, V., & Pal, S. (2018). A survey of classification techniques in student dropout prediction. International Journal of Computer Science and Engineering, 10(1), 41-49.
7. García, L., López, S., & Sánchez, M. (2020). Machine learning for student dropout prediction: A Naive Bayes approach. Journal of Educational Data Mining, 12(3), 22-35.
8. García, S., Luengo, J., & Herrera, F. (2018). Data preprocessing in data mining. Springer.
9. García-Sánchez, J.-N. (2020). Predicting student dropout risk using logistic regression: A case study. Journal of Educational Research and Practice, 10(3), 45-60.
10. He, J., & Xu, D. (2019). Application of multilayer perceptron in predicting student dropout risk in online learning environments. Educational Technology Research & Development, 67(4), 879-892.

11. Kaur, P., & Singh, S. (2021). A comparative analysis of machine learning algorithms for student dropout prediction. Journal of Educational Technology Systems, 49(3), 369-384.
12. Kemper, L., Vorhoff, G., & Wigger, B. U. (2020). Predicting student dropout: A machine learning approach. European Journal of Higher Education, 10(1), 28-47.
13. Liu, S., Xie, L., & Huang, X. (2019). A random forest approach for predicting student dropout in higher education. Journal of Educational Technology & Society, 22(3), 74-85.
14. Liu, X., & Wu, J. (2017). Predicting student dropout using decision trees. Proceedings of the 2017 International Conference on Educational Data Mining, 90-98.
15. Martins, M. V., Tolledo, D., Machado, J., Baptista, L. M., & Realinho, V. (2021). Early prediction of student's performance in higher education: a case study. In Trends and Applications in Information Systems and Technologies: Volume 1 9 (pp. 166-175). Springer International Publishing.
16. Ndou, N., Ajoodha, R., & Jadhav, A. (2020, November). Educational data-mining to determine student success at higher education institutions. In 2020 2nd International Multidisciplinary Information Technology and Engineering Conference (IMITEC) (pp. 1-8). IEEE.
17. Olmedo-Cifuentes, I., & Martínez-León, I. M. (2022). University dropout intention: analysis during covid-19. Journal of Management and Business Education, 5(2), 97-117.
18. Panagiotakopoulos, T., Kotsiantis, S., Kostopoulos, G., Iatrellis, O., & Kameas, A. (2021). Early dropout prediction in MOOCs through supervised learning and hyperparameter optimization. Electronics, 10(14), 1701.
19. Paura, L., & Arhipova, I. (2014). Cause Analysis of Students' Dropout Rate in Higher Education Study Program. Procedia - Social and Behavioral Sciences, 109, 1282–1286.
20. Prenkaj, B., Velardi, P., Stilo, G., Distante, D., & Faralli, S. (2020). A survey of machine learning approaches for student dropout prediction in online courses. ACM Computing Surveys (CSUR), 53(3), 1-34.
21. Rodríguez, P., Villanueva, A., Dombrovskaia, L., & Valenzuela, J. P. (2023). A methodology to design, develop, and evaluate machine learning models for predicting dropout in school systems: the case of Chile. Education and Information Technologies, 28(8), 10103-10149.
22. Sarker, I. H., Hossain, M. S., & Kabir, M. R. (2021). Prediction of student dropout using machine learning models. International Journal of Artificial Intelligence in Education, 31(4), 607-623.
23. Segura, M., Mello, J., & Hernández, A. (2022). Machine learning prediction of university student dropout: Does preference play a key role?. Mathematics, 10(18), 3359.

24. Srivastava, N., Hinton, G. E., & Krizhevsky, A. (2014). Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15(1), 1929-1958.
25. Tena, M. T., & Henson, R. K. (2016). Analyzing student dropout in higher education using decision trees. Journal of Educational Data Mining, 8(2), 55-70.
26. Tharwat, A. (2016). Decision tree classifiers: A comprehensive review. International Journal of Computer Science & Information Technology, 8(1), 33-42.
27. Tinto, V. (1993). Leaving college: Rethinking the causes and cures of student attrition (2nd ed.). University of Chicago Press.
28. Yaacob, W. W., Sobri, N. M., Nasir, S. M., Norshahidi, N. D., & Husin, W. W. (2020, March). Predicting student drop-out in higher institution using data mining techniques. In Journal of Physics: Conference Series (Vol. 1496, No. 1, p. 012005). IOP Publishing.
29. Yağcı, M. (2022). Educational data mining: prediction of students' academic performance using machine learning algorithms. Smart Learning Environments, 9(1), 11.
30. Zhou, L., Zhang, L., & Wu, J. (2020). Predicting student dropout using random forest algorithms: A case study in online education. International Journal of Educational Technology, 21(2), 128-139.