Online-Journals.org

iJOE International Journal of Online and Biomedical Engineering

iJOE | eISSN: 2626-8493 | Vol. 20 No. 3 (2024) | OPEN ACCESS

https://doi.org/10.3991/ijoe.v20i03.46287

PAPER

Educational Data Mining: Employing Machine Learning Techniques and Hyperparameter Optimization to Improve Students' Academic Performance

Mohamed Bellaj(✉), **Ahmed Ben Dahmane**, **Said Boudra**, **Mohammed Lamarti Sefian**

Higher Normal School (ENS), Abdelmalek Essaadi University, Tetouan, Morocco

mohamed.bellaj@etu.uae.ac.ma

ABSTRACT

Educational data mining (EDM) is a specialized field within data mining that focuses on extracting valuable insights from academic data across high school and university levels. A common practice in EDM involves predicting students' grades to identify at-risk individuals and improve the efficiency of academic tasks. This knowledge benefits students, parents, and institutions equally. Early detection enables interventions that improve student performance. The literature presents various prediction strategies, each with its own unique advantages and disadvantages. This study aims to comprehensively evaluate the methods, tools, and applications of machine learning (ML) and data mining (DM) in education. The main goal is to improve the accuracy of predicting academic achievements by employing eight widely recognized ML algorithms: naïve bayes (NB), k-nearest neighbors (KNN), support vector machine (SVM), random forest (RF), logistic regression (LR), extreme gradient boost (XGBOOST), and ensemble voting classifier (EVC). The focus is on improving data quality by eliminating instances of noise. Performance evaluation involves assessing parameters such as accuracy, precision, F-measure, and recall. Incorporating cross-validation and hyperparameter tuning improves classification accuracy. The ML models outperform other ensemble approaches, providing a valuable tool for predicting student performance and assisting educators in making proactive decisions through timely alerts.

KEYWORDS

educational data mining (EDM), machine learning (ML) algorithms, students' performance, hyper-parameter tuning, cross-validation

1 INTRODUCTION

One of the most challenging and popular research topics in educational data mining (EDM) is student performance modeling [1]. Academic achievement is

Bellaj, M., Dahmane, A.B., Boudra, S., Sefian, M.L. (2024). Educational Data Mining: Employing Machine Learning Techniques and Hyperparameter Optimization to Improve Students' Academic Performance. *International Journal of Online and Biomedical Engineering (iJOE)*, 20(3), pp. 55–74. https://doi.org/10.3991/ijoe.v20i03.46287

Article submitted 2023-10-27. Revision uploaded 2023-12-03. Final acceptance 2023-12-03.

influenced by a variety of variables in non-linear ways. The researchers found EDM more appealing because of the abundance of educational datasets. EDM is a field of study where techniques from data mining (DM) are utilized to enhance and predict students' academic performance [2].

For example, utilizing the collected data can assist teachers in enhancing their capacity to comprehend students' learning gaps and develop instructional strategies. Students can use it to enhance their learning experiences and broaden their educational pursuits [3]. It also affects how the administrator makes decisions that will result in high-quality output [4]. EDM utilizes computational techniques to analyze and display educational data. This test may be used to predict a student's performance and identify their strengths and weaknesses in skills and knowledge. It may be used to identify the behaviors of undesirable pupils and provide them with advice. These models can aid instructors in recruiting learners, obtaining feedback, and creating courses. Data about education is gathered from a variety of sources, including surveys, heuristic shops, and the Internet. A few DM strategies are employed to address educational challenges, and EDM can use a variety of DM techniques. For example, classification is the most effective strategy known for building educational prediction models. Within the realm of classification methodologies, which include constructs such as random forest (RF), logistic regression (LR), DT, naïve bayes (NB) trees, support vector machine (SVM), and k-nearest neighbors (KNN), a variety of techniques can be identified [5].

This paper introduces an innovative model for evaluating student performance, integrating a distinctive set of attributes known as "behavioral and academic features." The dataset used in this study is obtained from an e-learning platform called Kalboard 360 [4]. Utilizing a variety of DM techniques, our model meticulously analyzes these datasets with the goal of understanding the influence of students' behavioral attributes on their academic achievements. Furthermore, we aim to improve our understanding of the inherent characteristics of these specific features through an extended data collection phase and more meticulous preprocessing methodologies. This demonstrates how EDM could provide significant assistance to an intelligent decision support system. Based on the system's performance, educational institutions and teachers, in particular, can receive assistance in making judgments regarding a wide range of topics, such as potential dropouts, pass/fail ratios, grading and evaluation methods, participation levels, and learning styles.

The primary focus of this study is to address a significant challenge in an e-learning environment, specifically the task of predicting students' performance by analyzing their engagement and participation data gathered through an e-learning tool. Consequently, this paper introduces how a comprehensive ML approach that utilizes ensemble methods can be used to forecast student performance and facilitate informed decisions aimed at enhancing their academic achievements. This paper is organized as follows: Section 2 provides a summary of the background on DM algorithms and related tools relevant to our research. Section 3 delves into the strategy used and the different phases of our process. In Section 4, we provide an explanation of the ML algorithms used in our analytical work. Section 5 presents the results we have obtained. Section 6 delves deeper into discussing the results we have achieved and provides interpretations. Finally, in Section 7, we present the conclusions of our analytical study and outline directions for future research.

2 RELATED WORKS

In a pertinent study, predictive models for student performance were developed and compared [6]. This research involved integrating demographic information, academic history, and behavioral indicators, including the involvement of both students and parents in the educational process. In addition to evaluating the NB and artificial neural network (ANN) algorithms, the study also examined a variety of ensemble methods, including bagging, boosting, and RF. Notably, the ANN model exhibited superior performance compared to other data mining techniques, with the boosting ensemble method emerging as the most effective approach.

The primary goal of [7] was to predict the academic qualifications of students. To accomplish this goal, the researchers evaluated a sample dataset consisting of 152 representative high school students using five different classification algorithms and four methods for attribute assessment. Remarkably, among the various approaches explored, the multilayer perceptron (MLP) emerged as the most effective classifier.

The authors of a second comparative study on the effectiveness of EDM approaches [8] analyzed four methods for identifying students who are most likely to encounter difficulties in introductory programming courses at an early stage. They also highlighted the importance of data preprocessing and algorithm optimization in improving the efficiency of these methods. The optimized SVM performed the best overall in the end.

Within the trio of research objectives explored in the study conducted by [9], significant emphasis was placed on predicting students' performance upon completion of their degree program. To accomplish this, relevant data, such as students' academic achievements throughout the entire four-year program, as well as information regarding their pre-admission grades, were considered. The resulting graduation grade, which was divided into five potential groups (A, B, C, D, and E), was the main variable of interest in the study. In pursuit of this goal, the authors meticulously employed a variety of classification techniques, including KNN, NB, neural networks, and RF. Notably, the NB method achieved the highest level of accuracy, reaching 83.65%.

In the study [10], various classification algorithms were utilized to analyze an online course database. The objective was to identify the most effective model for categorizing academic achievement based on key attributes. The classifiers assessed included DT, KNN, SVM, and ANN. To evaluate the efficiency of these algorithms, a range of performance metrics, including precision, accuracy, F-measure, and recall, were utilized. Notably, DT and ANN yielded the most favorable results.

A precise prediction of students' academic achievement requires a thorough understanding of the characteristics and factors that influence student performance and success [11]. In pursuit of this objective, [12] meticulously scrutinized 357 academic papers on student performance, examining the impact of 29 distinct factors. Predominantly focused on psychomotor skills, these attributes included various factors such as academic background, past achievements, student involvement, and demographic characteristics such as gender, high school performance, and self-discipline. It is worth noting that variables such as student motivation, behaviors, socio-economic challenges, academic stagnation, and vocational transitions significantly influence dropout rates.

In a recent study [13], the effectiveness of seven algorithms was examined, including deep learning, ensemble methods, classical SVM, and KNN. While random forest achieved an accuracy level of around 91%, its applicability to various online learning platforms was found to be limited.

In another study [14], three DT algorithms—RT, rep tree, and J48—were used to predict students' academic performance. Cross-validation was utilized to evaluate the performance of the prediction model. Based on the data, it was found that RT outperformed the other algorithms with an accuracy rate of 75.188%.

Furthermore, as shown in [15], an increase in prediction accuracy was achieved by using parameter tuning. In comparison to traditional machine learning algorithms trained with the under-sampling technique, LR achieved 75%, MLP achieved 76%, RF achieved 75%, and KNN achieved 73%. [16] exhibited enhanced results: LR achieved 89.7%, MLP achieved 86.5%, NB achieved 78.4%, and RF achieved 88.8%. Similarly, for oversampling, LR achieved 78%, MLP achieved 64%, RF achieved 50%, and KNN achieved 55%.

Utilizing the same dataset extracted from Kalboard 360, the initial investigation [17] uses five traditional ML algorithms, supplemented by four ensemble techniques: bagging, boosting, stacking, and voting. The individual model F1 scores are as follows: DT (0.675), RF (0.777), GBT (0.714), NB (0.654), and KNN (0.664). Remarkably, using ensemble approaches leads to a significant improvement in model performance. The second study [18] explores the three phases of big data analytics (descriptive, predictive, and prescriptive analytics) and their advantages. The study employs a case study that uses ML techniques (DTs and RF) to predict student performance and offers recommendations and guidance for improving student outcomes. The goal is to encourage educational institutions to utilize their data to enhance student performance and overall benefits.

In summary, numerous studies have explored addressing educational challenges through the use of DM methods. Nonetheless, only a limited number of these research efforts have integrated additional techniques, such as hyperparameter optimization, into the learning process. This inclusion aims to improve the accuracy of the obtained outcomes and produce more favorable results. This study will utilize various ML algorithms to predict learner performance. Additionally, hyperparameter optimization will be used to improve the accuracy of the final results.

3 METHODOLOGY

In this study, we formulated a strategy for creating predictive models through DM by using the cross-industry standard process for data mining (CRISP-DM) model [19]. The CRISP-DM framework serves as a methodological and procedural guide for DM, offering a comprehensive structure for project implementation. Widely acknowledged as a prominent methodology in the field of DM [20], the CRISP-DM process (see Figure 1) has garnered substantial recognition, as evidenced by a 2014 online survey conducted by KDNuggets, a prominent global community in the field of data mining.

The authors of [20] conducted a comprehensive examination of the application of the CRISP-DM process model, and their findings highlighted its widespread adoption across various domains. Of particular significance is its frequent use within the education sector, where it has been notably and extensively implemented more than in any other industry [21].

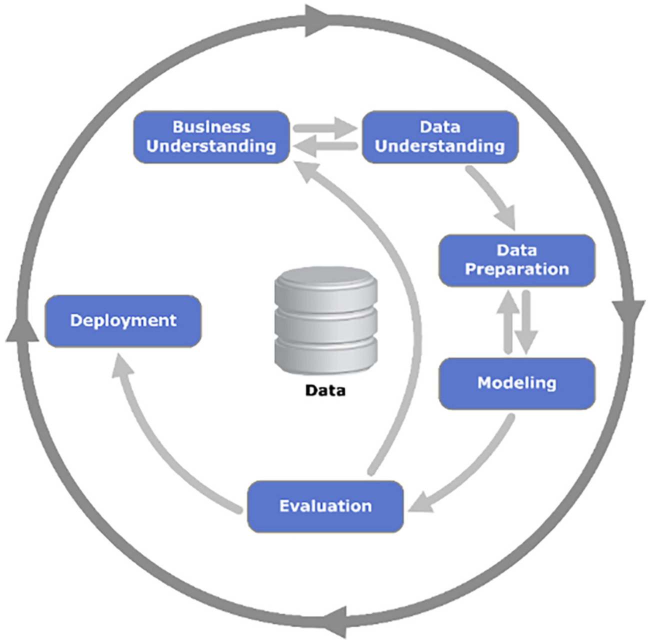

Fig. 1. Phases of the original CRISP-DM reference model [22]

3.1 Business understanding

The initial phase of a data mining project is crucial, as it involves understanding the project's goals and requirements from a business perspective. This understanding forms the foundation for defining the DM problem and creating an initial strategy to achieve the desired objectives. Essentially, the initial phase of the project involves understanding the project's goals and requirements and then using that knowledge to define the scope of the DM challenge and devise an initial strategy to achieve the set objectives.

3.2 Data understanding

The data understanding phase, which constitutes the second stage of the CRISP-DM framework, plays a crucial role in carefully selecting the data for analysis, assessing the quality of the existing data, and establishing a connection between the data and its significance in relation to business objectives. Focused primarily on uncovering hidden trends and patterns, this phase aims to detect possible inconsistencies, guide the analytical approach, and define subcategories that will later undergo hypothesis testing.

3.3 Data preparation

This phase involves all the essential tasks necessary to construct the final dataset from the initial raw data, preparing it for analysis. It is likely that the data preparation tasks will go through multiple iterations, often following a non-linear sequence. These tasks involve selecting appropriate tables, datasets, and attribute values and then transforming, refining, and structuring the data to ensure it is suitable for subsequent stages of analysis.

3.4 Modeling

A set of modeling approaches is selected and utilized in this step. Their parameters are thus adjusted to optimal values. This stage involves the use of various statistical techniques or ML algorithms, such as regression, classification, clustering, and recommendation. The data must adhere to specific requirements for certain procedures. As a result, it is often necessary to revisit the data preparation stage [22].

3.5 Evaluation

A model or models have been developed at this stage of the project that appear suitable for use in data analysis. The processes involved in constructing the model must be thoroughly evaluated and verified before completion to ensure that the prototype is suitable for the intended mission. Ensuring that all important issues have been effectively addressed is a crucial goal. The decision on how to utilize the findings from the extraction should mark the conclusion of this step. The choice of assessment metrics depends entirely on the project's requirements, the algorithm used, the desired outcomes, and so on.

3.6 Deployment

The deployment could involve generating reports or implementing the data mining process after the model has been developed, validated, and evaluated using test and validation data.

4 MACHINE LEARNING ALGORITHMS

4.1 Business understanding

In traditional classroom settings, educators and learners can interact directly, enabling teachers to gain insights from facial expressions, body language, and vocal cues. This enhances their understanding of students' learning experiences. This interaction also enables the customization of instructional methods to meet the individual needs of students. Similarly, e-learning systems have the potential to facilitate teacher-student communication. Many modern scholars acknowledge the significant potential of using data mining techniques to optimize the utilization of educational data. Notably, student behavior is a crucial factor in predicting academic performance.

In this paper, we conduct parallel experiments to develop a predictive model for student academic performance. Anticipated advancements in predicting academic outcomes in the education sector offer the potential to deliver multifaceted benefits for both students and educators. By using educational statistics, our aim is to demonstrate these benefits and create a continuous channel of communication between teachers and students, potentially enhancing learning environments.

Our methodology involved identifying a subset of attributes that were expected to directly impact students' grades, thereby improving the accuracy of predictions. To accomplish this, we created graphs (see Figure 2) illustrating the relationship between these attributes and the grades achieved by the students, followed by a comprehensive analysis. Subsequently, we utilized various supervised learning algorithms to classify students into three distinct grade categories.

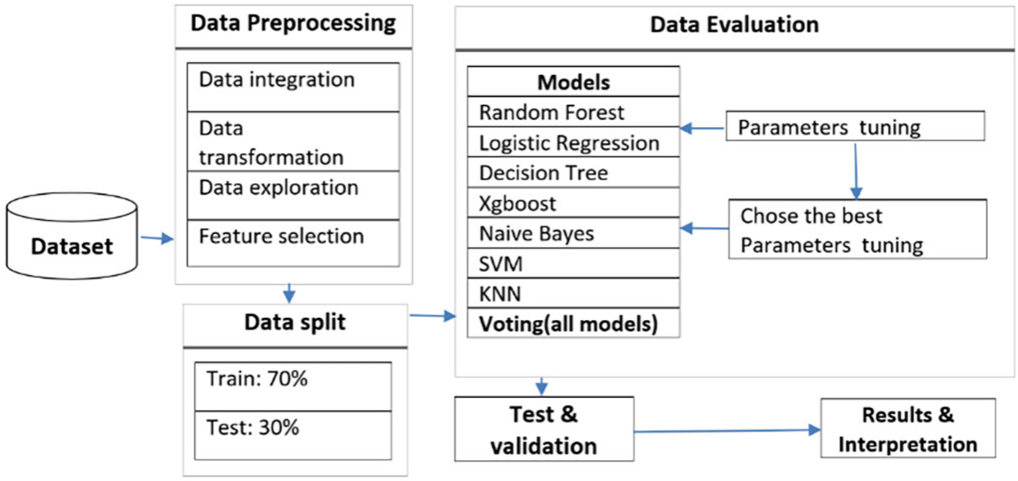

Fig. 2. Research steps for the student performance prediction model

4.2 Data understanding

The dataset used in this study was obtained from the Kalboard 360 learning management system, which is implemented in junior high schools and includes academic data. To enhance the learning process, a multi-agent junior high school called Kalboard 360 was established [23]. This innovative technology allows simultaneous access to educational content through internet-connected devices. Data is collected using the Experience API, a tool for tracking student activities (xAPI). The learning architecture incorporates xAPI, which enables the monitoring of learning progress. Through the use of the Experience API, providers can define students, activities, and objects that encapsulate the learning experience.

Table 1. Student features and their description

Feature Category	Feature	Description
Demographical Features	Gender	The gender of the student (female or male)
	Nationality	Student nationality
	Relation	Student's contact parent such as (father or mum)
Academic Background Features	StageID	Stage student belongs such as (Lower level, Middle level, and high level)
	GradeID	Grade student belongs such as (G-01, G-02, G-03, G-04, G-05, G-06, G-07, G-08, G-09, G-10, G-11, G-12)
	SectionID	Section student belongs such as (A, B, C).
	Topic	Course topic such as (Math, English, IT, Arabic, Science, Quran)
	Student Absence Days	Student absence days (Above-7, Under-7)
	Semester	School year semester as (First or second)

(Continued)

Table 1. Student features and their description *(Continued)*

Feature Category	Feature	Description
Behavioral Features	Raised Hands	How many times the student raises his/her hand on classroom (numeric: 0-100)
	Visited Resources	How many times the student visits a course content (numeric: 0-100)
	Announcements View	How many times the student checks the new announcements (numeric: 0-100)
	Discussion	How many times the student participated on discussion groups (numeric: 0-100)
Parent Participation in learning process	Parent School Satisfaction	This feature obtains the Degree of parent satisfaction from school as follow (Good, Bad)
	Parent Answering Survey	Parent is answering the surveys that provided from school or not (Yes, No).
Performance Level	Class	Total score (Low-Level: 0 to 69, Middle-Level: 70 to 89, High: interval 90 to 100)

The dataset comprises 480 instances, each associated with 16 attributes. These attributes are grouped into three categories: (1) demographic features, including gender and nationality; (2) academic background features, encompassing education, socioeconomic status, and grade level; and (3) social attributes, such as participation in class discussions, access to educational materials (in the sciences), responsiveness to parental surveys, and parental satisfaction with the school. The dataset is distributed as follows: 127 instances for the low class, 211 for the medium class, and 142 for the high class (see Table 1).

4.3 Data preparation

After collecting the data, we apply several pre-processing techniques to enhance the quality of the dataset. Data pre-processing is an essential step in the knowledge discovery process, involving data cleaning, data reduction, data transformation, and feature selection. During the preparation phase, we inspect the dataset to ensure that there are no unwanted or unnecessary values. This process is called cleaning. After completing the cleaning process, we reviewed the data and eliminated any fields that were unnecessary or irrelevant to the study's purpose. The cleaned data must be converted into a suitable format (transformation) to enable the more effective functioning of ML algorithms (such as feature selection).

a) Label encoding

In the field of ML, it is common to come across datasets containing multiple labels in one or more columns. These labels can be in the form of characters or numeric values. However, this data cannot be directly input into a ML model in its raw format. To make the data understandable for the model, a technique called label encoding is often used. Label encoding involves converting labels into a numeric format, enabling their integration into a ML model. This preprocessing step is crucial for supervised learning methods.

In label encoding, each value within a categorical column is typically replaced with a corresponding number ranging from 0 to N-1. The Label Encoder is a utility class that facilitates the normalization of labels.

b) Feature selection

This mechanism involves the careful selection of a pertinent subset of attributes that can effectively represent data while excluding redundant or insignificant elements. This intentional curation enhances the quality of data, which in turn improves the performance of the learning algorithm.

4.4 Modeling

During this stage, a thoughtful selection of modeling methods, procedures, or their synergistic combinations is made. This is followed by determining the optimal parameter values for the selected algorithms. Typically, there are a variety of potential modeling approaches to choose from for a given task. Some methods can be customized to address specific data quality constraints or data formats. As a result, this process often involves iterative iterations until the desired quality benchmarks for the model are achieved [24].

a) Random forest

Random forest and DT methodologies are closely related [25]. DTs are known for their high variability and low bias. By aggregating multiple DTs in the RF ensemble, the variance aspect of the model is mitigated. The process of averaging predictions in a RF helps in generating predictions for unknown samples.

$$I = \frac{1}{N}\sum_{N=1}^{N} \int(x) \tag{1}$$

N *represents the total number of individual DT in the ensemble, while prediction* ***i*** *corresponds to the estimate made by the i-th DT.*

The RF technique performs data analysis by using a collection of DT, combining predictions from each tree, and then determining the best course of action. Furthermore, RF operates based on an ensemble learning paradigm, utilizing the bagging algorithm, and it has the capability to handle datasets that contain missing values [26].

b) Logistic regression

Logistic regression stands as a widely used linear classification algorithm that is adept at modeling the probability of a binary outcome (1 or 0) based on one or multiple predictor variables. The logistic regression equation takes the form of the logistic function, also known as the sigmoid function, and is used to model a linear combination of the predictor variables [27].

The formulation for LR is typically represented as follows:

$$\mathrm{P} = (\mathrm{Y} = 1 \mid \mathrm{X}) = \frac{1}{1 + e^{-(\beta 0 + \beta 1 X1 + \beta 2 X2 + \cdots + \beta p X p)}} \tag{2}$$

P(Y = 1 | X) *represents the probability of the binary outcome being 1, given the predictor variables 1,2, …, X1,X2, …, Xp.*

β0 + β1X1 + β2X2 + … + βpXp *are the coefficients associated with each predictor variable.*

1, 2, …, X1,X2, …, Xp *are the predictor variables.*

e *is the base of the natural logarithm.*

The logistic function is designed to map any input value (denoted as z) to a range spanning from 0 to 1. This crucial characteristic enables the LR to infer the likelihood of the binary outcome being 1, based on the linear combination of predictor variables.

c) Decision tree

The DT algorithm uses a tree-like model to analyze potential outcomes, including incident results [28]. In this tree model, the target variables have a discrete range of values. Within the tree structure, branches and leaves represent relationships between features that correspond to class labels. The entropy equation is presented below:

$$\text{Entropy}(S) = -\sum_{i=1}^{c} pi\ log_2(pi) \tag{3}$$

s *represents the group of data points at the bulge.*
c *number of unique class labels in the dataset.*
pi *the proportion of data points in set S that belong to class i.*
log$_2$ *the base-2 logarithm.*

The tree structure, characterized by various nodes and edges, provides a visual representation of the interaction among variables. DTs excel when there is a monotonic progression among attributes. However, they face challenges when dealing with linear relationships, which could lead to instability. The comprehensibility of a complex decision tree, especially when there are numerous terminal nodes, can be quite challenging.

d) Naïve bayes

The NB classifier is a widely used classification algorithm that follows the principles of Bayes' theorem through mathematical principles [29].

$$P\left(B|A\right) = \frac{P(B\,|\,A)P(A)}{P(B)} \tag{4}$$

Bayes' theorem explains the relationship between a given class variable y and a dependent feature vector comprising attributes x1 to xj.

One notable advantage of the NB Classifier is its efficiency in terms of computational time, outperforming other machine learning algorithms in this respect. It excels at handling categorical input variables as opposed to numerical ones. However, a drawback is that it assumes all features to be independent variables, which poses practical implementation challenges [30].

e) Support vector machines

Support vector machines classifies data by identifying the hyperplane that maximizes the separation between classes in the training dataset [31]. Mathematically, this hyperplane is defined as follows:

$$\int(x) = a^T x + c \tag{5}$$

a represents the dimensional coefficient.
c represents the offset.

One of the key advantages of SVM is its flexibility to choose from a variety of kernels, which allows for the handling of complex, structured datasets through different kernel options. Furthermore, SVM showed reduced susceptibility to overfitting. However, choosing the right kernel can be challenging due to its crucial role. Conversely, when dealing with larger datasets, SVM may require significant processing time [32].

f) K-nearest neighbor

The methodology used in KNN involves examining K instances within the dataset that are close to the observed data point. Subsequently, the algorithm uses the results to predict the variable y of the observed instance [33]. The Euclidean distance serves as the metric for measuring the proximity of two observations. The equation for Euclidean distance is as follows:

$$d(x_i, y_i) = \sqrt{(x_{i,1} - y_{i,1})^2 + \ldots. + (x_{i,m} - y_{i,m})^2} \tag{6}$$

The KNN approach is notable for its minimal computational demands, as it does not require an initial training phase and learns directly from the dataset during prediction generation. Its simplicity of implementation is attributed to the requirement of only two factors: the K value and the distance function. However, challenges arise when dealing with extensive datasets, and the algorithm's effectiveness diminishes when applied to high-dimensional data [34].

g) Extreme gradient boosting

It constitutes a modern addition to the collection of ML algorithms. Rooted in the concept of gradient boosting, XGBoost seamlessly integrates principles from both the fields of ML and optimization [35, 36]. The objective function of the XGBoost algorithm is defined as follows:

$$\mathrm{O(t)} = \sum_{i=0}^{n} Q\left(y_{i,} y^{n-1} + \int_{i} (x_i) \right) + K \quad (7)$$

Then normalization function:

$$Nor(f_t) = 0{,}5\lambda \sum_{i=0}^{T} W_j^2 + kT \quad (8)$$

κ *represents the controlling factor for the leaf node number.*
T *represents the leaf node number.*
W_j *represents the weightage of the j leaf nodes.*
λ *represents the overfitting controlling factor.*
K *represents a constant.*

This algorithm showcases proficiency in handling datasets of different sizes, including both small and large datasets. However, challenges may arise when the dataset contains a large number of categorical values.

h) Ensemble voting classifiers

Ensemble vote classifier involve the amalgamation of multiple ML models during the training phase. The final prediction is made by selecting the session with the highest probability, which is determined by the majority of votes from the component models [37–39]. There are two primary categories of voting classifiers:

Ensemble voting classifier-hard (EVCH): In this approach, each individual classifier contributes its vote towards the intended output class, resulting in the selection of the class that has received the highest number of votes, mathematically.

$$H_v = mod\left\{ C_1(x), C_2(x), C_3(x), \ldots, C_n(x) \right\} \quad (9)$$

Where, ()nCx = Output class from classifier

Ensemble voting classifier-soft (EVCS): Each classifier generates a probability associated with the output class. These probabilities are then assigned weights and combined, taking into account the respective significance of each classifier. Eventually, the class with the highest combined probability is identified as the output. Mathematically, this procedure can be illustrated as follows:

$$S_v = argmax_i \sum_{j=1}^{u} H_j P_{ij} \;\; i \in \{O,1\}, [j = 1, 2, 3, \ldots, n] \quad (10)$$

Where:
H_j = *heap up to jth classifier*
P_{ij} = *Probabilities originating by the algorithm*

Ensemble methods provide a more comprehensive perspective on data, helping to mitigate bias and variability in predictions across different scenarios. However, interpreting ensemble methods can be challenging at times because of the presence of multiple classifiers within the framework.

i) Hyper tuning

The process of selecting the appropriate hyperparameters for a ML system is commonly known as hyperparameter tuning or optimization. Hyperparameters are parameters utilized to regulate the learning process. To effectively handle various patterns in the data, a specific ML model may require different configurations for constraints, weights, or learning rates [40]. These adjustments, known as hyperparameters, are crucial for enabling the model to effectively tackle the specific machine learning task at hand. In the following section, the hyper-tuning approach will be used to achieve optimal results from the mentioned ML algorithms.

j) Cross-validation

Cross-validation aims to validate the accuracy of numerical results when evaluating hypothesized relationships within the dataset. In this study, a 10-fold cross-validation technique was used to partition the dataset. This partitioning was carried out using Scikit-Learn's model selection function [41], specifically utilizing the stratified K-Fold subfunction for cross-validation. The performance scores of the machine learning classifiers were evaluated using the cross-validation score and GridSearchCV subfunctions.

5 RESULTS AND EVALUATION

5.1 Environment

The experiments were conducted on a personal computer equipped with 8GB of RAM and 4 Intel cores, each operating at 2.30GHz. The entire project was developed using Python Notebook version 3.9.13. Toolkits, such as Pandas, NumPy, Matplotlib, and Scikit-Learn [42, 43], were utilized to assess the proposed classification models and make comparisons. Furthermore, a 10-fold cross-validation technique was used to divide the dataset into training and testing subsets.

5.2 Performance metric

Following an extensive exploration of various machine learning algorithms, we conducted comprehensive simulations. As a result, predictive models were used to classify students into low, medium, and high-performance categories. A diverse range of performance metrics, such as accuracy, precision, sensitivity, specificity, F1-score, and the area under the receiver operating characteristic (ROC-AUC) curve, were calculated and assessed. The mathematical formulations for each metric are provided below, with TP representing true positive, TN representing true negative, FP representing false positive, and FN representing false negative.

$$Accuracy = \frac{TP + TN}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN} + \mathrm{TN}} \tag{11}$$

$$Precision = \frac{TP}{\mathrm{TP} + \mathrm{FP}} \tag{12}$$

$$Recall = \frac{TP}{\mathrm{TP} + \mathrm{FN}} \tag{13}$$

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{14}$$

5.3 Feature selection

In our analysis, we used four different classifiers (DT, LR, RF, and XGBoost) to determine the best prediction outcome. We systematically evaluated all the classifiers to determine the most favorable result. In the first iteration, the Random Forest classifier produced the best result. Through subsequent iterations, the Random Forest classifier consistently outperformed the others across all metrics. It's worth noting that the classifiers were ranked according to their accuracy levels (refer to Table 2).

Table 2. Features selection using classification techniques results

	F1	Accuracy	Precision	Recall	Roc_Auc
RF	0.841	0.841	0.847	0.841	0.94
LR	0.758	0.758	0.755	0.758	0.88
DT	0.750	0.750	0.753	0.750	0.82
GB	0.716	0.716	0.712	0.716	0.91

As indicated in Table 2, our analysis has shown that the RF classifier demonstrated superior performance.

Given the binary nature of the classification task and the presence of imbalanced data, our primary focus is on evaluating the F1 and ROC AUC scores. The F1 score, which represents the harmonic mean of precision and recall, and the ROC AUC score are of paramount importance. The differentiation between these categories is crucial because misapplied marketing strategies can have significant financial consequences for users.

The optimal set of five features (see Figure 3), referred to as "5Feat," identified through this procedure includes visited resources, raised hands, student absences, announcements viewed, and discussions.

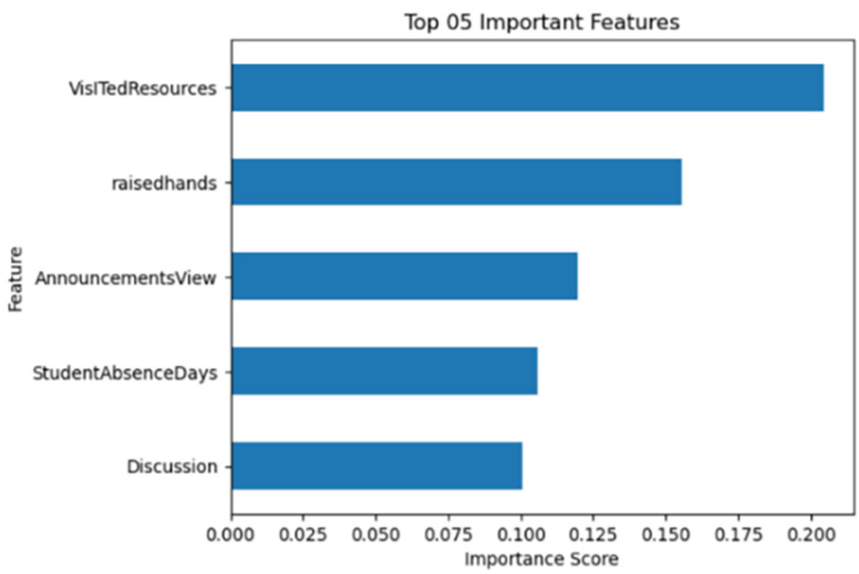

Fig. 3. The Five important features

6 DISCUSSION

6.1 Evaluating results using base classifiers

When forecasting student performance, numerous factors can influence the predictive model. Within this study, we considered behavioral attributes to be crucial factors capable of influencing students' academic achievements. In Table 3, we demonstrate the impact of the 5Feat features using various classifiers, including RF, LR, NB, DT, KNN, SVM, XGBOOST, and EVC. For each classifier, we categorized the classification outcomes into two groups: "classification results with 5Feat" and "classification results without 5Feat."

Table 3. Classification techniques results with and without the 5Feat features selected

	Without (5Feat)					With (5Feat)				
	F1	Accuracy	Precision	Recall	Roc_Auc	F1	Accuracy	Precision	Recall	Roc_Auc
Voting	0.59	0.59	0.59	0.59	0.74	0.84	0.84	0.84	0.84	0.95
RF	0.57	0.57	0.57	0.57	0.73	0.82	0.82	0.82	0.82	0.95
XGB	0.57	0.57	0.57	0.57	0.73	0.80	0.80	0.80	0.80	0.93
DT	0.56	0.56	0.57	0.56	0.67	0.78	0.78	0.78	0.78	0.84
KNN	0.48	0.48	0.47	0.48	0.67	0.64	0.64	0.64	0.64	0.84
LR	0.47	0.47	0.47	0.47	0.68	0.72	0.72	0.72	0.72	0.89
NB	0.44	0.44	0.44	0.44	0.69	0.78	0.78	0.79	0.78	0.92
SVM	0.44	0.44	0.19	0.44	0.64	0.67	0.67	0.68	0.67	0.88

Table 3 presents the outcomes achieved by all classifiers with and without the optimally selected features. It is evident that the Voting Classifier model outperforms other data mining techniques. Specifically, the Voting Classifier model achieved an accuracy of 84% when using the 5Feat set, while it achieved an accuracy of 59% without including behavioral features. An accuracy of 84% implies that 403 out of 480 students are accurately categorized into their respective group tags (low, medium, and high), while 77 students are misclassified.

In terms of the recall and precision metrics, the achieved results were 84% when using the 5Feat feature and 59% without it. A recall and precision value of 84% indicates that 380 students are correctly classified out of the total count of both correctly and incorrectly categorized cases. Regarding the F-measure, the results are 0.84% with 5Feat and 0.59% without behavioral features.

The experimental findings underscore the significant influence of learner behavior on students' academic performance. By training the dataset using ensemble parameters, improved accuracy can be achieved.

6.2 Hyperparameter optimization

To ensure accurate predictions, it is crucial to optimize the hyperparameters of the machine learning model. In our proposed approach, we utilized GridsearchCV to fine-tune the hyperparameters [41]. To enhance precision and prevent model overfitting, we utilized the hyperparameter optimization technique to carefully select an optimal set of hyperparameters for the learning algorithm. We performed tenfold cross-validation to train and evaluate our models, comprehensively assessing algorithm performance using relevant metrics.

6.3 Results of evaluation using ensemble methods

In this section, we employed ensemble methods to enhance the accuracy of the evaluated outcomes by combining traditional classifiers such as RF, LR, NB, DT, KNN, SVM, and XGB.

Table 4. Classification techniques results using HPO

	Without (HPO)					With (HPO)				
	F1	Accuracy	Precision	Recall	Roc_Auc	F1	Accuracy	Precision	Recall	Roc_Auc
Voting	0.84	0.84	0.84	0.84	0.95	0.87	0.86	0.87	0.87	0.95
RF	0.82	0.82	0.82	0.82	0.95	0.85	0.84	0.84	0.86	0.94
XGB	0.80	0.80	0.80	0.80	0.93	0.85	0.84	0.84	0.86	0.94
DT	0.78	0.78	0.78	0.78	0.84	0.80	0.79	0.79	0.80	0.87
KNN	0.64	0.64	0.64	0.64	0.84	0.71	0.69	0.69	0.72	0.89
LR	0.72	0.72	0.72	0.72	0.89	0.83	0.81	0.81	0.83	0.93
NB	0.78	0.78	0.79	0.78	0.92	0.80	0.78	0.79	0.81	0.92
SVM	0.67	0.67	0.68	0.67	0.88	0.82	0.80	0.80	0.82	0.93

As shown in Table 4, significant improvements are observed in the results when ensemble methods with hyperparameter optimization (HPO) are applied. These enhanced results are then combined using a voting process to achieve the best predictive performance for students. By using the voting method, we achieve an impressive accuracy of 86%. These levels of accuracy indicate that 412 students are correctly categorized into the appropriate labels, while 68 students are misclassified. The improved results of the voting algorithm are presented in Table 4. This underscores the improvement of our prediction model by integrating ensemble methods and refining parameters.

6.4 Confusion matrix

After implementing our proposed hyperparameter tuning technique, the ML models were trained with an emphasis on reducing bias and addressing the risks of overfitting. Subsequently, to ensure the integrity of the results and reduce variance, the models underwent cross-validation testing, which is designed to prevent data leakage. Table 5 presents the confusion matrices for all the classifiers.

Table 5. Results of the algorithms utilized in the study's confusion matrix

		Predicted		
		L	*M*	*H*
Confusion Matrix (RF)				
Actual	*L*	0.22	0.014	0
	M	0.028	0.32	0.076
	H	0	0.042	0.31
Confusion Matrix (LR)				
Actual	*L*	0.22	0.014	0
	M	0.014	0.33	0.076
	H	0	0.083	0.26
Confusion Matrix (NB)				
Actual	*L*	0.22	0.0069	0
	M	0.028	0.26	0.13
	H	0	0.056	0.29
Confusion Matrix (DT)				
Actual	*L*	0.19	0.035	0
	M	0.042	0.31	0.076
	H	0	0.056	0.29
Confusion Matrix (KNN)				
Actual	*L*	0.2	0.028	0
	M	0.056	0.28	0.09
	H	0	0.13	0.22
Confusion Matrix (SVM)				
Actual	*L*	0.22	0.014	0
	M	0.014	0.33	0.083
	H	0	0.09	0.26
Confusion Matrix (XGB)				
Actual	*L*	0.22	0.0069	0
	M	0.035	0.32	0.069
	H	0	0.049	0.3
Confusion Matrix (Voting)				
Actual	*L*	0.21	0.021	0
	M	0.021	0.33	0.076
	H	0	0.021	0.33

6.5 Deployment

In this study, employment was implemented as an analytical web service that relies on prediction rules, with a specific focus on identifying potential failures. After making predictions, feedback on the predictions and their processing is used to generate new data, which in turn improves the model's predictive capabilities.

7 CONCLUSION

The EDM approach to predicting student performance is successful because it helps identify students at risk of failing due to their below-average performance. Early detection can help take necessary actions and ultimately prevent students from being dropped or failing. The substantial amount of research being conducted in this field is evidence of the significance of performance prediction. These methodologies mainly involve different stages of the data mining process, utilize various classification algorithms, and demonstrate their findings by applying their approach to different datasets. Such approaches can have significant value for educational institutions, benefiting students and parents. However, it is equally important that the performance prediction technique used is reliable and has undergone rigorous analysis. Flawed steps in a process can lead to misleading results and inaccurate conclusions. In this study, we meticulously examined feature selection techniques and carefully curated a refined dataset. We utilized multiple prediction models and found that the ensemble learning technique achieved the highest overall accuracy of 86%. Furthermore, our research emphasized the significant enhancements attained through the integration of cross-validation and hyperparameter tuning into the proposed algorithms.

Moving forward, we aim to expand this research by subjecting it to rigorous evaluation across a wider range of datasets. Furthermore, we aim to integrate deep learning into the prediction process, potentially using a hybrid approach. This may involve exploring alternative methods for selecting features and thoroughly examining the resulting outcomes.

8 REFERENCES

[1] M. K. Gupta and Pravin Chandra, "A comprehensive survey of data mining," *International Journal of Information Technology*, vol. 12, no. 4, pp. 1243–1257, 2020. https://doi.org/10.1007/s41870-020-00427-7

[2] A. Namoun and Abdullah Alshanqiti, "Predicting student performance using data mining and learning analytics techniques: A systematic literature review," *Applied Sciences*, vol. 11, no. 1, p. 237, 2020. https://doi.org/10.3390/app11010237

[3] R. Siva Kumar, "Effects of social media on academic performance of the students," *The Online Journal of Distance Education and eLearning*, vol. 8, no. 2, pp. 90–97, 2020.

[4] C. Romero and S. Ventura, "Educational data mining: A review of the state of the art," *IEEE Transactions on Systems, Man, and Cybernetics*, Part C, vol. 40, no. 6, pp. 601–618, 2010. https://doi.org/10.1109/TSMCC.2010.2053532

[5] S. Bharara, S. Sabitha, and Bansal, "Application of learning analytics using clustering data mining for students' disposition analysis," *Education and Information Technologies*, vol. 23, no. 2, pp. 957–984, 2018. https://doi.org/10.1007/s10639-017-9645-7

[6] A. M. Shahiri, W. A. Husain, and N. A. Rashid, "A review on predicting student's performance using data mining techniques," *Procedia Computer Science*, vol. 72, pp. 414–422, 2015. https://doi.org/10.1016/j.procs.2015.12.157

[7] E. A. Amrieh, T. Hamtini, and I. Aljarah, "Mining educational data to predict student's academic performance using ensemble methods," *Int. J. Database Theory Appl.*, vol. 9, pp. 119–136, 2016.

[8] P. Kaur, M. Singh, and G. S. Josan, "Classification and prediction based data mining algorithms to predict slow learners in education sector," *Procedia Comput. Sci.*, vol. 57, pp. 500–508, 2015. https://doi.org/10.1016/j.procs.2015.07.372

[9] E. B. Costa, B. Fonseca, M. A. Santana, F. F. de Araújo, and J. Rego, "Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses," *Comput. Hum. Behav.*, vol. 73, pp. 247–256, 2017. https://doi.org/10.1016/j.chb.2017.01.047

[10] R. Asif, A. Merceron, S. A. Ali, and N. G. Haider, "Analyzing undergraduate students' performance using educational data mining," *Comput. Educ.*, vol. 113, pp. 177–194, 2017.

[11] E. Al Nagi and N. Al-Madi, "Predicting students performance in online courses using classification techniques," in *Proceedings of the 2020 International Conference on Intelligent Data Science Technologies and Applications (IDSTA)*, *IEEE*, Valencia, Spain, 2020, pp. 51–58. https://doi.org/10.1109/IDSTA50958.2020.9264113

[12] A. Alshanqiti and A. Namoun, "Predicting student performance and its influential factors using hybrid regression and multi-label classification," *IEEE Access*, vol. 8, pp. 203827–203844, 2020. https://doi.org/10.1109/ACCESS.2020.3036572

[13] A. Hellas, P. Ihantola, A. Petersen, V. V. Ajanovski, M. Gutica, T. Hynninen, A. Knutas, J. Leinonen, C. Messom, and S. N. Liao, "Predicting academic performance: A systematic literature review," in *Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*, 2018, pp. 175–199. https://doi.org/10.1145/3293881.3295783

[14] M. Adnan, A. Habib, J. Ashraf, S. Mussadiq, A. A. Raza, M. Abid, M. Bashir, and S. U. Khan, "Predicting atisk students at different percentages of course length for early intervention using machine learning models," *IEEE Access*, vol. 9, pp. 7519–7539, 2021.

[15] E. C. Abana, "A decision tree approach for predicting student grades in research project using Weka," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 7, pp. 285–289, 2019.

[16] N. Mduma, K. Kalegele, and D. Machuve, "Machine learning approach for reducing students dropout rates," *International Journal of Advanced Computer* Research, vol. 9, no. 42, pp. 156–69, 2019. https://doi.org/10.19101/IJACR.2018.839045

[17] F. Saleem, Z. Ullah, B. Fakieh, and F. Kateb, "Intelligent decision support system for predicting student's E-learning performance using ensemble machine learning," *Mathematics*, vol. 9, no. 17, 2021. https://doi.org/10.3390/math9172078

[18] M. L. Boughouas, Y. Kissoum, A. Mouhssen, M. A. Karek, and S. Mazouzi, "Towards big educational data analytics," in *International Conference on Advanced Aspects of Software Engineering (ICAASE)*, 2022, pp. 1–6. https://doi.org/10.1109/ICAASE56196.2022.9931565

[19] C. Shearer, "The CRISP-DM model: The new blueprint for data mining," *J. Data Warehous.*, vol. 5, no. 4, pp. 13–22, 2000.

[20] S. Faure, "Descriptive analytics, machine learning, and deep learning viewed via the lens of CRISP-DM," Retrieved from https://www.kdnuggets.com/2018/05/descriptive-analytics-machine-learning-deep-learning-crisp-dm.html

[21] C. Catley, K. Smith, C. McGregor, and M. Tracy, "Extending CRISP-DM to incorporate temporal data mining of multidimensional medical data streams: A neonatal intensive care unit case study," in *22nd IEEE International Symposium on Computer-Based Medical Systems*, Albuquerque, NM, USA, 2009, pp. 1–5. https://doi.org/10.1109/CBMS.2009.5255394

[22] C. Schröer, F. Kruse, and J. M. Gómez, "A systematic literature review on applying CRISPDM process model," *Procedia Comput. Sci.*, vol. 181, pp. 526–534, 2021. https://doi.org/10.1016/j.procs.2021.01.199

[23] T. A. Eppes, I. Milanovic, and K. Wright, "Improving student readiness for inquiry-based learning: An engineering case study," *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 16, no. 1, pp. 4–17, 2020. https://doi.org/10.3991/ijoe.v16i01.12051

[24] E. A. Amrieh, T. Hamtini, and I. Aljarah, "Preprocessing and analyzing educational data set using X-API for improving student's performance," in *IEEE Jordan Conf. Appl. Electr. Eng. Comput. Technol. (AEECT)*, Amman, Jordan, 2015, pp. 1–5. https://doi.org/10.1109/AEECT.2015.7360581

[25] I. El Guabassi, Z. Bousalem, R. Marah, and A. Qazdar, "Comparative analysis of supervised machine learning algorithms to build a predictive model for evaluating students' performance," *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 17, no. 2, pp. 90–105, 2021. https://doi.org/10.3991/ijoe.v17i02.20025

[26] Nan Jiang, Fei Fu, Hua Zuo, Xiuping Zheng, and Qinghe Zheng, "A Municipal PM2.5 forecasting method based on random forest and WRF model," *Engineering Letters*, vol. 28, no. 2, pp. 312–321, 2020.

[27] K. Berková, D. Frendlovská, M. Chalupová, A. Kubišová, K. Krpálková Krelová, and D. Kolářová, "International comparison of higher education representatives' and students' attitudes towards feedback learning," *International Journal of Engineering Pedagogy (iJEP)*, vol. 13, no. 4, pp. 141–157, 2023. https://doi.org/10.3991/ijep.v13i4.37017

[28] H. Pallathadka, A. Wenda, E. Ramirez-Asís, M. Asís-López, J. Flores-Albornoz, and K. Phasinam, "Classification and prediction of student performance data using various machine learning algorithms," *Materials Today: Proceedings*, vol. 80, pp. 3782–3785, 2023. https://doi.org/10.1016/j.matpr.2021.07.382

[29] C. Gomes and L. S. Almeida, "Advocating the broad use of the decision tree method in education," *Practical Assessment, Research, and Evaluation*, vol. 22, no. 1, p. 10, 2017.

[30] K. Sunday, P. Ocheja, S. Hussain, S. S. Oyelere, B. O. Samson, and F. J. Agbo, "Analyzing student performance in programming education using classification techniques," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 15, no. 2, pp. 127–144, 2020. https://doi.org/10.3991/ijet.v15i02.11527

[31] Perez, Joann Galopo, and Eugene S. Perez, "Predicting student program completion using Naïve Bayes classification algorithm," *International Journal of Modern Education and Computer Science*, vol. 13, no. 3, pp. 57–67, 2021. https://doi.org/10.5815/ijmecs.2021.03.05

[32] J. Hou, "Online teaching quality evaluation model based on support vector machine and decision tree," *Journal of Intelligent & Fuzzy Systems*, vol. 40, no. 2, pp. 2193–2203, 2021. https://doi.org/10.3233/JIFS-189218

[33] A. Dirin and C. A. Saballe, "Machine learning models to predict students' study path selection," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 16, no. 1, pp. 158–183, 2022. https://doi.org/10.3991/ijim.v16i01.20121

[34] Yuliato, Lili Dwi, Agung Triayudi, and Ira Diana Sholihati, "Implementation educational data mining for analysis of student performance prediction with comparison of K-Nearest neighbor data mining method and decision tree C4. 5," *Jurnal Mantik*, vol. 4, no. 1, pp. 441–451, 2020.

[35] R. M. Mathew and R. Gunasundari, "A review on handling multiclass imbalanced data classification in education domain," in *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, 2021, pp. 752–755.

[36] Asseman, Amal, Mohamed Khaldi, and Souhaib Aammou, "Enhancing the prediction of student performance based on the machine learning XGBoost algorithm," *Interactive Learning Environments*, vol. 31, no. 6, pp. 3360–3379, 2023.

[37] Wang, Chixiang, and Junqi Guo, "A data-driven framework for learners' cognitive load detection using ECG-PPG physiological feature fusion and XGBoost classification," *Procedia Computer Science*, vol. 147, pp. 338–348, 2019. https://doi.org/10.1016/j.procs.2019.01.234

[38] Jaw, Ebrima, and Xueming Wang, "Feature selection and ensemble-based intrusion detection system: An efficient and comprehensive approach," *Symmetry*, vol. 13, no. 10, p. 1764, 2021. https://doi.org/10.3390/sym13101764

[39] Khatri, Ajay, Shweta Agrawal, and Jyotir M. Chatterjee, "Wheat seed classification: Utilizing ensemble machine learning approach," *Scientific Programming*. https://doi.org/10.1155/2022/2626868

[40] R. Vidhya and G. Vadivu, "Towards developing an ensemble based two-level student classification model (ESCM) using advanced learning patterns and analytics," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 7, pp. 7095–7105, 2021. https://doi.org/10.1007/s12652-020-02375-3

[41] A. Candelieri, I. Giordani, F. Archetti, K. Barkalov, I. Meyerov, A. Polovinkin, A. Sysoyev, and N. Zolotykh, "Tuning hyperparameters of a SVM-based water demand forecasting system through parallel global optimization," *Comput. Oper. Res.*, vol. 106, pp. 202–209, 2019. https://doi.org/10.1016/j.cor.2018.01.013

[42] Y. Ariyanto, B. Harijanto, and A. N. Asri, "Analyzing student's learning interests in the implementation of blended learning using data mining," *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 16, no. 11, pp. 153–160, 2020. https://doi.org/10.3991/ijoe.v16i11.16453

[43] M. M. J. Douglass, "Book review: Hands-on machine learning with Scikit-Learn, Keras, and Tensorflow," 2nd Edition by Aurélien Géron, *Phys. Eng. Sci. Med.*, vol. 43, pp. 1135–1136, 2019. https://doi.org/10.1007/s13246-020-00913-z

9 AUTHORS

Mohamed Bellaj is a doctoral student at the Higher Normal School (ENS), Abdelmalek Essaadi University, Tetouan, Morroco (E-mail: mohamed.bellaj@etu.uae.ac.ma).

Ahmed Ben Dahmane is a Professor at the Higher Normal School (ENS), Abdelmalek Essaadi University, Tetouan, Morroco (E-mail: abendahman@uae.ac.ma).

Said Boudra, Higher Normal School (ENS), Abdelmalek Essaadi University, Tetouan, Morroco (E-mail: ssboudra@gmail.com).

Mohammed Lamarti Sefian is a Professor at the Higher Normal School (ENS), Abdelmalek Essaadi University, Tetouan, Morroco (E-mail: lamarti.mohammed.sefian@uae.ac.ma).