Effective Feature Prediction Models for Student Performance

Bashayer Alsubhi
Department of Computer Science and Artificial Intelligence, College of Computer Science and Engineering, University of Jeddah, Saudi Arabia
balsubhe.stu@uj.edu.sa

Basma Alharbi
Department of Computer Science and Artificial Intelligence, College of Computer Science and Engineering, University of Jeddah, Saudi Arabia
bmalharbi@uj.edu.sa

Nahla Aljojo
College of Computer Science and Engineering, Department of Information System and Technology, University of Jeddah, Saudi Arabia
nmaljojo@uj.edu.sa (corresponding author)

Ameen Banjar
College of Computer Science and Engineering, Department of Information System and Technology, University of Jeddah, Saudi Arabia
abanjar@uj.edu.sa

Araek Tashkandi
College of Computer Science and Engineering, Department of Information System and Technology, University of Jeddah, Saudi Arabia
astashkandi@uj.edu.sa

Abdullah Alghoson
College of Computer Science and Engineering, Department of Information System and Technology, University of Jeddah, Saudi Arabia
alghoson@uj.edu.sa

Anas Al-Tirawi
College of Engineering, Computing and Design, Department of Computer Science, Dar Al-Hekma University, Saudi Arabia
atirawi@dah.edu.sa

Received: 30 August 2023 | Revised: 15 September 2023 | Accepted: 18 September 2023

 DOI: https://doi.org/10.48084/etasr.6345

ABSTRACT

The ability to accurately predict how students will perform has a significant impact on the teaching and learning process, as it can inform the instructor to devote extra attention to a particular student or group of students, which in turn prevents those students from failing a certain course. When it comes to educational data mining, the accuracy and explainability of predictions are of equal importance. Accuracy refers to the degree to which the predicted value was accurate, and explainability refers to the degree to which the predicted value could be understood. This study used machine learning to predict the features that best contribute to the performance of a student, using a dataset collected from a public university in Jeddah, Saudi Arabia. Experimental analysis was carried out with Black-Box (BB) and White-Box (WB)

machine-learning classification models. In BB classification models, a decision (or class) is often predicted with limited explainability on why this decision was made, while in WB classification models decisions made are fully interpretable to the stakeholders. The results showed that these BB models performed similarly in terms of accuracy and recall whether the classifiers attempted to predict an A or an F grade. When comparing the classifiers' accuracy in making predictions on B grade, the Support Vector Machine (SVM) was found to be superior to Naïve Bayes (NB). However, the recall results were quite similar except for the K-Nearest Neighbor (KNN) classifier. When predicting grades C and D, RF had the best accuracy and NB the worst. RF had the best recall when predicting a C grade, while NB had the lowest. When predicting a D grade, SVM had the best recall performance, while NB had the lowest.

Keywords-student performance; artificial neural networks; support vector machine; Naïve Bayes; k-nearest neighbor

I. INTRODUCTION

Machine Learning (ML) models enable machines to make predictions. In general, ML models are categorized into supervised, unsupervised, and reinforcement models. Many studies have demonstrated how to successfully involve ML models to solve problems involving predictions [1-19]. The learning processes are different from category to category. Models can learn by providing the input data and its corresponding output (supervised learning), providing only input data (unsupervised learning), or providing only rewards (reinforcement learning). In many cases, ML models contain hundreds of nodes that are connected to solve a given problem. It is difficult for humans to understand the intuition behind the predictions made. For this, an ANN is often denoted as a Black-Box (BB) model, where it is difficult for domain experts to "see" inside it and "understand" why it is making certain decisions. This lack of interpretability of the model and the explainability of the decision causes a trust issue. Deep learning is another common example of BB models, as it is difficult for a human to understand the millions or billions of calculations made by such algorithms. For that, some domains, such as healthcare and the military, enforce regulations and constraints when using ML to make important decisions.

According to the Statement on Algorithmic Transparency and Accountability [20], providing explanations of the algorithm's decision-making process is becoming mandatory. This is because some algorithms can lead to harmful bias, which can have a legal or financial impact or incorrect predictions. Fortunately, this encouraged the development of methods that are used to explain BB models. This growing area of research is often known as explainable AI (XAI), which aims to develop methods that explain AI systems to their stakeholders [17]. This is important when these systems are too complex or ambiguous and encompasses the underlying causes of the system's methods or procedures to provide information that helps various stakeholders to better understand these systems. In XAI, explainable methods are classified into global or local, where global methods explain the prediction of all instances and local methods explain the prediction for a particular instance or group of instances [15]. On the other hand, some models are inherently interpretable because there is no need for additional steps to explain their behavior. This category of models is often denoted as White-Box (WB) models [19]. Such models include Decision Trees (DT), Rule-Based (RB) systems, Contrast Patterns (CP), and Fuzzy Patterns (FP). Often, these WB models provide a trade-off between accuracy and explainability, meaning that training BB models will often provide better accuracy than WB. Due to this assumption, researchers seek to use BB models by adding a layer of explainability on top of them. However, in [21], it was stated that state-of-the-art WB models can achieve prediction performance comparable to BB models, and their use is encouraged instead of explaining BB models. This study aimed to train state-of-the-art WB models in the educational data mining domain and systematically compare their performance against BB models. Educational Data Mining (EDM) aims to develop methods that study and explore educational data that could be from face-to-face education, e-learning and Learning Management Systems (LMSs) or Intelligent Tutoring Systems (ITSs), and Adaptive Educational Hypermedia Systems (AEHSs) [17]. After analyzing educational data, EDM tries to evaluate the educational systems to improve the learning processes and better understand learners and learning.

In [1], four data mining techniques, ANN, DT, SVM, and Naïve Bayes (NB), were used on a dataset from Princess Norah University with a total of 4,078 students who took the General Aptitude Test (GAT) and the Scholastic Achievement Admission Test (SAAT). The four classifiers were compared for their accuracy, precision, recall, and F1-score, showing that ANN was the best in terms of accuracy (79%) and precision (81%), while DT was the best in terms of recall (80%) and F1-score (81%) and the NB classifier exhibited the worst performance. In [4], regression algorithms were used on a dataset of 85 students and 3 student feature classes: personal features, educational features, and behavioral features. In [3], a recommendation system used RF, DT, and Linear Regression (LR) to maintain the best behavior for the proposed system. In [11], exploratory factor analysis, multiple linear regressions, cluster analysis, and correlations were used to predict student academic performance. In [8], a nonlinear predictive model was proposed that can be explained using the SHAP game-theory-based framework. In [6], a warning system was proposed using Multi-View Genetic Programming (MVGP). In [10], a prediction model was developed that used Genetic Programming-Interpretable Classification Rule Mining (GP-ICRM) that was optimal and interpretable. In [12], a model was proposed that used a rule-based genetic programming algorithm for prediction, achieving good performance with 89% precision, 86.7% recall, 87.5 F1-score, and 89.9% ROC-score. In [2], several WB and BB models were used. This study aimed to use CORELS, which is an interpretable model, and compare its performance with several WB and BB models to test the claim that SOTA interpretable models can achieve prediction performance comparable to BB [21].

II. PRELIMINARIES

A. Problem Definition

This study aims to predict the success or failure of a student in a specific course. Let $x(j)$ be a $1 \times n$ vector of student j grades at selected courses, where n is the number of selected courses, m is the number of students in the dataset, and $j = (1, \ldots, m)$. When packing $x(j)$ row by row, an $m \times n$ matrix X of student-course grades is obtained. Let Xt denote student-course grades at time $\leq t$. Let $y(t+1)$ denote the student-course grade for a selected course at time $\geq t$, such that $y(t+1) \in \{0, 1\}$, where 0 indicates failing a course and 1 indicates passing it. The objective is to predict a student's performance in a new course, given his/her performance in previous courses, in the form of a function f that maps input Xt to output $y(t+1)$) as:

$$y(t+1) = f(Xt) \tag{1}$$

Predicting students' passing or failing a course is a binary classification problem since $y(t+1)$ can be 0 or 1. The input Xt, that is, student grades, can be numerical or categorical, where numerical grades range from 0 to 100, and categorical grades can either be binary (0 or 1) to indicate failing or passing a course, or discretized letter grades (A+, A, B+, B, C+, ...). This study tested the proposed model on selected courses from the dataset.

B. Data Description and Preprocessing

This study used a dataset collected from a public university located in Jeddah, Saudi Arabia, which contained student enrollment records, each having student ID, course ID, and the grade obtained. The dataset had a total of 250 students and 180 courses collected from 2015 to 2019. At first, the unnecessary features were removed from the dataset. Then, the maximum, minimum, mean, median, and standard deviation of the grades were calculated for each student, course, and teacher, as shown in Table II. The dataset was discretized, as shown in Table III, by converting the continuous data into categorical. Performance was evaluated in terms of accuracy, recall, precision, complexity matrix, and speed. Accuracy is the ratio of correct predictions to the total number of input samples (3), while the complexity metric measures the ratio between the number of classes and the number of rules (4) [14]. Recall and precision, as shown in (1) and (2), are similar to accuracy but are often used on unbalanced data. Lastly, the speed of the model measures how fast the model can be trained and tested.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{1}$$

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{2}$$

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \tag{3}$$

$$Complexity = \frac{l}{\sum_{i=1}^{r} k_r} \tag{4}$$

where l is the number of classes (2 in this case), r is the number of rules, and k is the number of features used in the i-th rule [14].

TABLE I. SYMBOL NOTATIONS

Notation	Description
S	Student
C	Course
T	Teacher
n	Number of courses
m	Number of Students

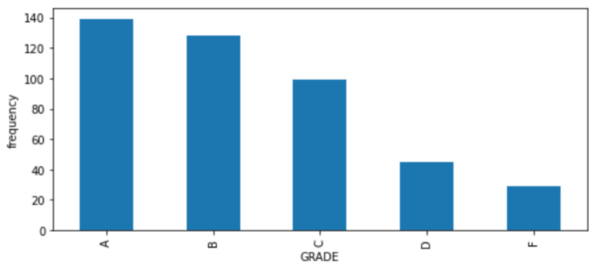

Fig. 1. Grade distribution.

TABLE II. DATASET STATISTICS ON THE GRADE OF EACH STUDENT, COURSE, AND TEACHER.

S avg	T avg	C avg	S med	T med	C med	S std	T std	C std	S min	T min	C min	S max	T max	C max
60.0	65.4	66.0	60.0	68.0	70.0	0.0	20.6	18.5	60.0	3.0	3.0	60.0	97.0	97.0
70.0	68.3	67.9	70.0	67.5	65.0	0.0	6.5	6.2	70.0	60.0	60.0	70.0	75.0	75.0
72.7	73.5	78.5	75.0	80.0	84.0	4.8	19.6	18.5	66.0	11.0	11.0	77.0	96.0	96.0
72.7	69.4	73.4	75.0	71.0	73.0	4.8	13.2	11.8	66.0	29.0	33.0	77.0	92.0	100.0
72.7	74.1	72.4	75.0	75.0	74.0	4.8	13.1	15.2	66.0	38.0	13.0	77.0	96.0	99.0

TABLE III. DATASET AFTER DISCRETIZATION

S avg	T avg	C avg	S med	T med	C med	S std	T std	C std	S min	T min	C min	S max	T max	C max
(59, 69]	(59, 69]	(59, 69]	(59, 69]	(59, 69]	(69, 79]	(-0.001, 2.944]	(16.214, 23.629]	(15.153, 18.543]	(59, 69]	(0, 59]	(0, 59]	(59, 69]	(89, 100]	(89, 100]
(69, 79]	(59, 69]	(59, 69]	(69, 79]	(59, 69]	(59, 69]	(-0.001, 2.944]	(4.443, 11.306]	(5.782, 11.819]	(69, 79]	(59, 69]	(59, 69]	(69, 79]	(69, 79]	(69, 79]
(69, 79]	(69, 79]	(69, 79]	(69, 79]	(79, 89]	(79, 89]	(2.944, 6.128]	(16.214, 23.629]	(15.153, 18.543]	(59, 69]	(0, 59]	(0, 59]	(69, 79]	(89, 100]	(89, 100]
(69, 79]	(69, 79]	(69, 79]	(69, 79]	(69, 79]	(69, 79]	(2.944, 6.128]	(11.306, 14.214]	(5.782, 11.819]	(59, 69]	(0, 59]	(0, 59]	(69, 79]	(89, 100]	(89, 100]
(69, 79]	(69, 79]	(69, 79]	(69, 79]	(69, 79]	(69, 79]	(2.944, 6.128]	(11.306, 14.214]	(11.819, 15.153]	(59, 69]	(0, 59]	(0, 59]	(69, 79]	(89, 100]	(89, 100]

TABLE IV. DATASET IN BINARY FORMAT

S AVG (0, 59]	S AVG (59, 69]	S AVG (69, 79]	S AVG (79, 89]	S AVG (89, 100]	T AVG (0, 59]	T AVG (59, 69]	...	T MAX (79, 89]	T MAX (89, 100]	C MAX (0, 59]	C MAX (59, 69]	C MAX (69, 79]	C MAX (79, 89]	C MAX (89, 100]
0	1	0	0	0	0	1	...	0	1	0	0	0	0	1
0	0	1	0	0	0	1	...	0	0	0	0	1	0	0
0	0	1	0	0	0	0	...	0	1	0	0	0	0	1
0	0	1	0	0	0	0	...	0	1	0	0	0	0	1
0	0	1	0	0	0	0	...	0	1	0	0	0	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
0	0	0	1	0	0	0	...	0	1	0	0	0	0	1
0	1	0	0	0	0	1	...	0	0	0	0	1	0	0
0	0	1	0	0	0	0	...	0	1	0	0	0	0	1
0	0	1	0	0	0	0	...	0	1	0	0	0	0	1
0	0	1	0	0	0	0	...	0	1	0	0	0	0	1

TABLE V. DATASET AFTER OVERSAMPLING

S AVG (0, 59]	S AVG (59, 69]	S AVG (69, 79]	S AVG (79, 89]	S AVG (89, 100]	T AVG (0, 59]	T AVG (59, 69]	...	T MAX (79, 89]	T MAX (89, 100]	C MAX (0, 59]	C MAX (59, 69]	C MAX (69, 79]	C MAX (79, 89]	C MAX (89,100]
0	1	0	0	0	0	1	...	0	1	0	0	0	0	1
0	0	1	0	0	0	1	...	0	0	0	0	1	0	0
0	0	1	0	0	0	0	...	0	1	0	0	0	0	1
0	0	1	0	0	0	0	...	0	1	0	0	0	0	1
0	0	1	0	0	0	0	...	0	1	0	0	0	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
0	1	0	0	0	0	1	...	0	1	0	0	0	0	1
1	0	0	0	0	0	1	...	0	1	0	0	0	0	1
1	0	0	0	0	0	0	...	0	1	0	0	0	0	1
1	0	0	0	0	0	1	...	0	1	0	0	0	0	1
1	0	0	0	0	0	0	...	0	1	0	0	0	0	1

III. METHODOLOGY

A. Framework Description

Figure 2 illustrates the framework. At first, the dataset was cleaned by removing unnecessary features and replacing all null values with zeros. Then, discretization was performed on the preprocessed dataset. After that, oversampling was used to equally split the dataset by every fold and obtain reasonable results from the unbalanced dataset. For every grade *y*, there is model training and testing, using the five-fold cross-validation technique. After cross-validation is performed, the average of every evaluation matrix is provided.

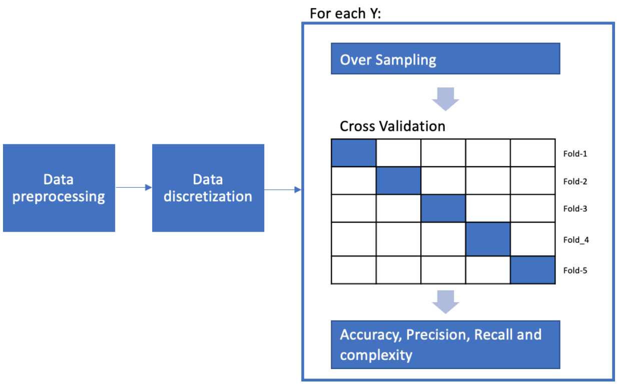

Fig. 2. Methodology framework.

B. Evaluation Metric

The 5-fold cross-validation was used to evaluate the model, where the dataset was split into five parts, and training and testing were performed five times. For each of the five iterations, one dataset part was used for testing and the others were used for training. Evaluation metrics were calculated in the testing set in each iteration i =1 kr, where l is the number of classes (2 in this case), r is the number of rules, and k is the number of features used in the i-th rule [14].

IV. RESULTS AND DISCUSSION

A. Experimental Settings

Jupyter Notebook was used as the development environment for Python3, along with Scikit-learn and Pandas for data analysis, Matplotlib for visualizations, and imodels for testing and evaluating the interpretable models. The tests were run on a MacBook Air laptop with a 1.1GHz quad-core Intel Core i5 CPU and 8 GB RAM.

B. Results

There are two types of ML models, WB and BB. WB models tend to be highly interpretable, meaning that it is easy for humans to understand the results provided, whereas BB models are not. The strength of WB models relies on finding biased results and preventing them from happening. This study selected CORELS as the main model, which is a state-of-the-art rule-based model that attempts to learn an optimal set of rules in each problem, and its performance was compared with several other WB and BB models. Standard Deviation (SD) measures the spread of the scores by the mean [22]. The results in Tables VI-XII have a very low SD after all the five folds, indicating good performance. As the dataset was unbalanced, the accuracy was 100% or close to it when predicting grades A and F for the CORELS, GreedyTree, C4.5, Bayesian Rule List (BRL), and Boosted Rules models without using the oversampling technique. In contrast, the Slipper model was not as good as the other models at predicting grade A. The accuracy in predicting grades B, C, and D is reasonably acceptable for all models.

Table VIII shows that when the model predicted grade A, GreedyTree and C4.5 classifiers outperformed CORELS by 0.02 and 0.03, respectively, in accuracy, while CORELS outperformed the Slipper classifier by 0.02 in accuracy and had the same accuracy with Bayesian and Boosted classifiers. The recall results were very close to each other. When the model predicted grades B, C, and D, the accuracy results of yhe GreedyTree, C4.5, Bayesian, and Boosted classifiers outperformed CORELS, while the Slipper classifier underperformed. In recall, the CORELS, Greedy Tree, C4.5, Bayesian, and Boosted classifiers had similar results, while the Slipper classifier had the lowest recall in predicting grade B. The recall of CORELS in predicting grade C was lower than the GreedyTree, C4.5, Bayesian, and Boosted classifier, but higher than the Slipper classifier. When predicting grade D, CORELS outperformed GreedyTree, C4.5, Slipper, and Boosted classifiers but underperformed the Bayesian classifier. For predicting grade F, the accuracy and recall of all WB classifiers were similar [23].

Tables VII and VIII show that when using oversampling, accuracy did not change dramatically, but recall improved. When predicting grade B with CORELS, the recall was 0.64 without and 0.96 with oversampling, which means that oversampling improved the model's testing performance. Table IX shows that CORELS took less training time than the C4.5, Bayesian, and Slipper classifiers. In addition, CORELS took more time to train than GreedyTree and Boosted classifiers, but had the least testing time among the rest WB classifiers. Table XI shows that the accuracy and recall results of these BB models were very similar when the classifiers predicted grades A and F. When the classifiers predicted grade B, SVM had the best accuracy results, while NB had the worst. On the other hand, the recall results are very close to each other except for the KNN classifier [24]. When the classifiers predicted grades C and D, RF had the best accuracy, and NB had the worst. When predicting grade C, RF had the best recall results, while NB had the worst. When predicting grade D, SVM had the best recall result and NB had the worst. Table XII shows that RF took the longest training time, while KNN took the shortest [25]. In addition, it is observed that RF took the longest testing time, while GB took the shortest. Moreover, when not using oversampling, Tables X and XI show that accuracy and recall were improved. When predicting grades A and F, WB and BB models had similar high accuracies. Moreover, WB models, except the Slipper classifier, had better accuracy than NB, SVM, and KNN when predicting grades B, C, and D. On the other hand, RF and GB models had similar accuracy as the WB models, except Slipper. Tables IX and XII show that C4.5, BRL, and Slipper models took longer time to process than BB models [26]. On the other hand, CORELS, BR, and GT took a similar time as the BB models. It can be stated that CORELS can provide high accuracy with minimal time.

TABLE VI. WB RESULTS

	Grade A			Grade B			Grade C			Grade D			Grade F		
Classifier	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**
CORELS	**1.0 (±0.00)**	**1.0 (±0.00)**	**1.0 (±0.00)**	**0.87 (±0.03)**	**0.75 (±0.07)**	**0.64 (±0.07)**	**0.80 (±0.02)**	**0.81 (±0.05)**	**0.42 (±0.03)**	0.8 (±0.04)	0.76 (±0.05)	0.60 (±0.09)	**0.95 (±0.01)**	**0.51 (±0.27)**	**0.49 (±0.00)**
Greedy Tree	**1.0 (±0.00)**	**1.0 (±0.00)**	**1.0 (±0.00)**	**0.87 (±0.04)**	**0.71 (±0.07)**	**0.67 (±0.05)**	**0.80 (±0.02)**	**0.65 (±0.09)**	**0.61 (±0.05)**	0.78 (±0.05)	0.68 (±0.06)	0.65 (±0.07)	0.93 (±0.01)	0.56 (±0.15)	0.47 (±0.18)
C4.5 Tree	**1.0 (±0.00)**	**1.0 (±0.00)**	**1.0 (±0.00)**	**0.87 (±0.04)**	**0.72 (±0.08)**	**0.65 (±0.14)**	0.79 (±0.02)	0.66 (±0.09)	0.58 (±0.08)	0.79 (±0.05)	0.67 (±0.06)	0.67 (±0.06)	0.94 (±0.02)	0.52 (±0.17)	0.47 (±0.18)
BRL	0.89 (±0.02)	0.47 (±0.06)	1.0 (±0.00)	0.77 (±0.03)	0.49 (±0.02)	1.0 (±0.00)	0.69 (±0.04)	0.48 (±0.03)	0.99 (±0.02)	0.74 (±0.07)	0.55 (±0.06)	0.99 (±0.01)	0.94 (±0.00)	0.51 (±0.10)	1.0 (±0.00)
Boosted Rules	0.89 (±0.02)	0.47 (±0.06)	1.0 (±0.00)	0.83 (±0.03)	0.90 (±0.13)	0.29 (±0.09)	0.79 (±0.02)	0.85 (±0.13)	0.37 (±0.15)	**0.83 (±0.02)**	**0.78 (±0.04)**	**0.64 (±0.08)**	0.94 (±0.01)	0.53 (±0.09)	0.96 (±0.01)
Slipper	0.79 (±0.22)	0.68 (±0.30)	0.37 (±0.22)	0.83 (±0.04)	0.81 (±0.17)	0.36 (±0.12)	0.73 (±0.11)	0.82 (±0.18)	0.30 (±0.15)	0.80 (±0.03)	0.8 (±0.10)	0.44 (±0.14)	0.94 (±0.02)	0.53 (±0.24)	0.49 (±0.29)

TABLE VII. WB RESULTS WITHOUT OVERSAMPLING

	Grade A			Grade B			Grade C			Grade D			Grade F		
Classifier	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**
CORELS	0.94 (±0.05)	0.74 (±0.24)	1.0 (±0.00)	0.83 (±0.06)	0.68 (±0.23)	0.96 (±0.04)	0.74 (±0.06)	0.68 (±0.19)	0.86 (±0.12)	0.76 (±0.05)	0.69 (±0.19)	0.92 (±0.08)	0.97 (±0.03)	0.80 (±0.21)	1.0 (±0.00)
Greedy Tree	0.96 (±0.01)	0.94 (±0.03)	0.99 (±0.02)	**0.90 (±0.02)**	**0.85 (±0.04)**	**0.96 (±0.02)**	0.86 (±0.03)	0.83 (±0.03)	0.90 (±0.06)	0.83 (±0.03)	0.81 (±0.03)	0.88 (±0.08)	**0.98 (±0.01)**	**0.97 (±0.01)**	**1.0 (±0.00)**
C4.5 Tree	**0.97 (±0.01)**	**0.94 (±0.02)**	**1.0 (±0.00)**	**0.90 (±0.01)**	**0.85 (±0.03)**	**0.97 (±0.02)**	**0.87 (±0.03)**	**0.82 (±0.02)**	**0.94 (±0.06)**	**0.84 (±0.03)**	**0.82 (±0.04)**	**0.88 (±0.08)**	**0.98 (±0.01)**	**0.97 (±0.01)**	**1.0 (±0.00)**
Bayesian Rule List	0.94 (±0.01)	0.89 (±0.02)	1.0 (±0.00)	0.85 (±0.03)	0.77 (±0.04)	1.0 (±0.00)	0.78 (±0.03)	0.70 (±0.03)	1.0 (±0.01)	0.81 (±0.05)	0.73 (±0.06)	0.99 (±0.01)	0.97 (±0.00)	0.94 (±0.00)	1.0 (±0.00)
Boosted Rules	0.94 (±0.01)	0.89 (±0.02)	1.0 (±0.00)	0.85 (±0.03)	0.78 (±0.04)	0.98 (±0.02)	0.78 (±0.02)	0.70 (±0.03)	0.96 (±0.04)	0.79 (±0.03)	0.80 (±0.08)	0.81 (±0.10)	0.96 (±0.01)	0.93 (±0.02)	1.0 (±0.00)
Slipper	0.92 (±0.03)	0.84 (±0.07)	1.0 (±0.00)	0.77 (±0.05)	0.79 (±0.06)	0.71 (±0.14)	0.69 (±0.05)	0.89 (±0.07)	0.43 (±0.11)	0.75 (±0.07)	0.88 (±0.03)	0.58 (±0.15)	0.92 (±0.09)	0.84 (±0.09)	1.0 (±0.00)

TABLE VIII. WB WITH OVERSAMPLING

	Time in sec.									
	Grade A		**Grade B**		**Grade C**		**Grade D**		**Grade F**	
Classifier	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
CORELS	0.0540 (±0.0053)	**0.0006 (±0.0001)**	0.0456 (±0.0058)	**0.0006 (±0.0002)**	0.0475 (±0.0047)	**0.0006 (±0.0003)**	0.0469 (±0.0108)	**0.0004 (±0.0001)**	1.6690 (±1.3231)	**0.0011 (±0.0009)**
Greedy Tree	**0.0036 (±0.0014)**	0.0023 (±0.0007)	**0.0073 (±0.0082)**	0.0026 (±0.0015)	**0.0042 (±0.0016)**	0.0022 (±0.0006)	**0.0038 (±0.2265)**	0.0024 (±0.0007)	**0.0035 (±0.0012)**	0.0024 (±0.0008)
C4.5 Tree	1.1081 (±0.1544)	0.0085 (±0.0035)	1.7746 (±0.2468)	0.0113 (±0.0038)	1.8600 (±0.1238)	0.0117 (±0.0010)	1.5420 (±0.2265)	0.0093 (±0.0007)	1.0215(±0.2051)	0.0077 (±0.0026)
BRL	18.4103 (±2.4006)	0.1596 (±0.0091)	19.4687 (±1.8279)	0.1336 (±0.0075)	1378.9996 (±2710.32)	0.1285 (±0.0099)	21.9118 (±2.7820)	0.1281 (±0.0124)	15.4696 (±1.8045)	0.1621 (±0.0163)
Boosted Rules	0.0137 (±0.0020)	0.0317 (±0.00324)	0.0148 (±0.0044)	0.0320 (±0.00199)	0.0161 (±0.0050)	0.0382 (±0.0063)	0.0614 (±0.0935)	0.0152 (±0.1190)	0.0160 (±0.0025)	0.0354 (±0.0050)
Slipper	146.633 (±215.272)	0.02814 (±0.0090)	152.67 (±220.28)	0.0229 (±0.0011)	37.1966 (±7.8914)	0.0253 (±0.0039)	150.98 (±217.499)	0.0243 (±0.0007)	42.8454 (±5.2960)	0.0207 (±0.0006)

TABLE IX. TRAIN AND TEST TIMES FOR THE WB MODELS

	Grade A			**Grade B**			**Grade C**			**Grade D**			**Grade F**		
Classifier	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**
NB	0.90 (±0.03)	0.49 (±0.08)	1.0 (±0.00)	0.65 (±0.16)	0.40 (±0.08)	0.82 (±0.06)	0.73 (±0.06)	0.55 (±±0.14	0.31)(±0.14)	0.78 (±0.05)	0.61 (±0.05)	0.77 (±0.11)	0.93 (±0.00)	0.48 (±0.09)	0.91 (±0.08)
SVM	**0.93 (±0.02)**	**0.71 (±0.37)**	**0.30 (±0.17)**	0.77 (±0.04)	0.0 (±0.00)	0.0 (±0.00)	0.75 (±0.05)	0.90 (±0.20)	0.14 (±0.05)	**0.80 (±0.04)**	**0.76 (±0.08)**	**0.56 (±0.11)**	0.93 (±0.03)	0.25 (±0.22)	0.15 (±0.13)
KNN	0.91 (±0.02)	0.59 (±0.33)	0.27 (±0.05)	0.78 (±0.05)	0.54 (±0.09)	0.45 (±0.09)	0.77 (±0.05)	0.66 (±0.04)	0.48 (±0.10)	**0.80 (±0.03)**	**0.70 (±0.03)**	**0.64 (±0.07)**	0.93 (±0.02)	0.37 (±0.22)	0.27 (±0.20)
Random Forest	0.91 (±0.03)	0.52 (±0.34)	0.35 (±0.19)	0.79 (±0.06)	0.54 (±0.05)	0.43 (±0.09)	0.80 (±0.02)	0.67 (±0.07)	0.60 (±0.05)	0.77 (±0.06)	0.63 (±0.09)	0.65 (±0.09)	**0.95 (±0.01)**	**0.51 (±0.27)**	**0.36 (±0.19)**
Gradient Boosting	0.90 (±0.03)	0.38 (±0.25)	0.63 (±0.41)	**0.83 (±0.03)**	**0.63 (±0.04)**	**0.62 (±0.06)**	**0.81 (±0.03)**	**0.71 (±0.07)**	**0.59 (±0.09)**	**0.80 (±0.04)**	**0.71 (±0.06)**	**0.67 (±0.07)**	0.91 (±0.02)	0.42 (±0.08)	1.0 (±0.00)

TABLE X. BB RESULTS WITHOUT OVERSAMPLING

	Grade A			**Grade B**			**Grade C**			**Grade D**			**Grade F**		
Classifier	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**	**Acc.**	**Prec.**	**Rec.**
NB	0.94 (±0.05)	0.74 (±0.23)	1.0 (±0.00)	0.74 (±0.10)	0.61 (±0.29)	0.90 (±0.03)	0.68 (±0.07)	0.71 (±0.21)	0.53 (±0.13)	0.77 (±0.04)	0.70 (±0.17)	0.81 (±0.09)	0.97 (±0.03)	0.80 (±0.21)	1.0 (±0.00)
SVM	0.94 (±0.05)	0.73 (±0.24)	1.0 (±0.00)	0.79 (±0.07)	0.65 (±0.25)	0.96 (±0.06)	0.71 (±0.05)	0.67 (±0.22)	0.80 (±0.09)	0.76 (±0.04)	0.68 (±0.19)	0.92 (±0.08)	0.96 (±0.04)	0.78 (±0.25)	1.0 (±0.00)
KNN	0.91 (±0.05)	0.71 (±0.27)	0.97 (±0.02)	0.76 (±0.05)	0.66 (±0.26)	0.88 (±0.08)	0.73 (±0.03)	0.67 (±0.21)	0.79 (±0.02)	0.79 (±0.06)	0.71 (±0.18)	0.86 (±0.04)	0.96 (±0.02)	0.81 (±0.20)	0.99 (±0.01)
Random Forest	**0.95 (±0.03)**	**0.80 (±0.19)**	**0.99 (±0.01)**	**0.88 (±0.07)**	**0.73 (±0.21)**	**0.96 (±0.03)**	**0.86 (±0.04)**	**0.78 (±0.14)**	**0.89 (±0.04)**	**0.83 (±0.07)**	**0.73 (±0.17)**	**0.86 (±0.07)**	**0.98 (±0.02)**	**0.84 (±0.18)**	**1.0 (±0.00)**
Gradient Boosting	0.94 (±0.04)	0.75 (±0.22)	0.99 (±0.01)	0.80 (±0.03)	0.69 (±0.23)	0.90 (±0.04)	0.78 (±0.07)	0.72 (±0.18)	0.85 (±0.07)	0.78 (±0.05)	0.71 (±0.19)	0.87 (±0.08)	0.97 (±0.03)	0.81 (±0.20)	1.0 (±0.00)

TABLE XI. BB RESULTS WITH OVERSAMPLING

	Time in sec.									
	Grade A		**Grade B**		**Grade C**		**Grade D**		**Grade F**	
Classifier	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
NB	0.0034 (±0.0020)	0.0024 (±0.0012)	0.0035 (±0.0021)	0.0024 (±0.0010)	0.0033 (±0.0013)	0.0026 (±0.0010)	0.0029 (±0.0013)	0.0025 (±0.0013)	0.0032 (±0.0014)	0.0026 (±0.0011)
SVM	0.0098 (±0.00618)	0.0046 (±0.0017)	0.0118 (±0.0047)	0.0065 (±0.0019)	0.0119 (±0.0047)	0.0065 (±0.0016)	0.0090 (±0.0037)	0.0050 (±0.0012)	0.0071 (±0.0029)	0.0040 (±0.0011)
KNN	0.0038 (±0.0037)	0.0076 (±0.0030)	0.0026 (±0.0006)	0.0067 (±0.0012)	0.0023 (±0.0003)	0.0056 (±0.0008)	0.0028 (±0.0015)	0.0068 (±0.0036)	0.0027 (±0.0012)	0.0081 (±0.0044)
Random Forest	0.1095 (±0.0128)	0.0076 (±0.0030)	0.1164 (±0.0129)	0.0114 (±0.0005)	0.1138 (±0.0091)	0.0109 (±0.0006)	0.1109 (±0.0084)	0.0110 (±0.0005)	0.1086 (±0.0119)	0.0113 (±0.0012)
Gradient Boosting	0.0390 (±0.0076)	0.0018 (±0.0003)	0.0373 (±0.0064)	0.0018 (±0.0001)	0.0376 (±0.0075)	0.0019 (±0.0002)	0.0388 (±0.0071)	0.0020 (±0.0003)	0.0452 (±0.0034)	0.0021 (±0.0001)

The largest complexity result indicates better interoperability. Table VI shows that CORELS and BRL are the most interpretable among the WB models. On the other hand, the GreedyTree classifier is the lowest interpretable model. Figure 3 shows that there is a positive relationship between complexity and classifier accuracy. When a classifier gets higher accuracy, the complexity gets better, except for the boosted classifier, which is fixed. Table IX shows that BRL and CORELS have the best complexity results among the other WB models.

TABLE XII. TRAIN AND TEST TIMES FOR THE BB MODELS

	Complexity				
	Grade A	**Grade B**	**Grade C**	**Grade D**	**Grade F**
CORELS	**1.0 (±0.0)**	**1.0 (±0.0)**	**0.8 (±0.2450)**	**1.0 (±0.0)**	**1.0 (±0.0)**
Greedy Tree	0.0075 (±0.0007)	0.0025 (±0.0003)	0.0029 (±0.0002)	0.0030 (±0.0003)	0.0140 (±0.0023)
C4.5 Tree	0.0267 (±0.0007)	0.0116 (±0.0012)	0.0102 (±0.0007)	0.0114 (±0.0010)	0.0464 (±0.0022)
BRL	**0.6599 (±0.2799)**	**0.5905 (±0.1524)**	**0.3078 (±0.0640)**	**0.3733 (±0.0326)**	**0.8333 (±0.2108)**
Boosted Rules	0.2 (±0.0)	0.2 (±0.0)	0.2 (±0.0)	0.2 (±0.0)	0.2 (±0.0)
Slipper	0.0990 (±0.0094)	0.06768 (±0.0087)	0.0804 (±0.0112)	0.0667 (±0.0098)	0.0905 (±0.0105)

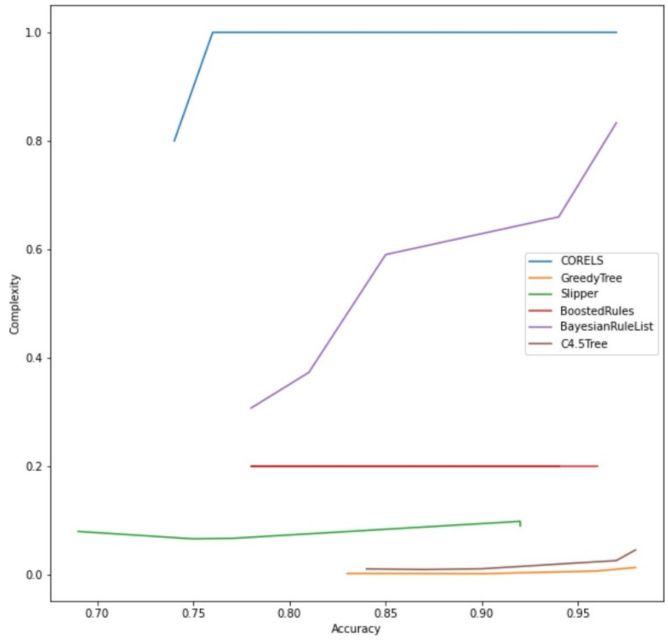

Fig. 3. WB models' complexity/accuracy relationship.

V. CONCLUSION

This paper compared the CORELS model with other WB and BB models. The results showed that CORELS outperformed the other WB and BB models. The dataset used in this study was obtained from a public institution in Jeddah, Saudi Arabia. In the future, this comparison is planned to be made on data from more colleges and other universities in Saudi Arabia and other countries. An effective prediction of how students will perform has a significant impact on both learning and teaching. This study offers a comprehensive examination of the efficacy of WB and BB categorization models in forecasting academic grades. While BB models provide decision-making procedures that are not easily understood, the emphasis on BB models brings clarity to the outcomes, which is essential for a wide range of stakeholders. The results of this study show that BB models consistently achieve high accuracy and recall rates, regardless of the anticipated grade category, whether it is an A or an F. Additionally, this study reveals significant variations in the precision and recall rates of particular classifiers while making predictions for grade B, with SVM outperforming NB. These results contribute to the understanding of the relative efficiency of various classifiers in educational settings. Furthermore, this study underscores a notable discrepancy in academic achievement when forecasting grades C or D. The RF algorithm had better accuracy compared to the others, whereas the NB offered the lowest level of effectiveness. These findings provide crucial information on the most reliable models for predicting grades within a specific range, which could provide insight to educational institutions aiming to enhance their classification systems. Furthermore, the data highlights the higher recall of RF in predicting grade C and the usefulness of SVM in predicting grade D. These distinctions offer useful information to schools seeking to improve their forecast accuracy for particular grade categories. This study provides practical insights that can be directly applied to improve grading prediction systems in similar educational situations, by utilizing actual educational data as the basis for analysis.

REFERENCES

[1] H. A. Mengash, "Using Data Mining Techniques to Predict Student Performance to Support Decision Making in University Admission Systems," *IEEE Access*, vol. 8, pp. 55462–55470, 2020, https://doi.org/10.1109/ACCESS.2020.2981905.

[2] S. E. Sorour, S. A. El Rahman, S. A. Kahouf, and T. Mine, "Understandable Prediction Models of Student Performance Using an Attribute Dictionary," in *Advances in Web-Based Learning–ICWL 2016: 15th International Conference*, Rome, Italy, Oct. 2016, pp. 161–171.

[3] A. Tarik, H. Aissa, and F. Yousef, "Artificial Intelligence and Machine Learning to Predict Student Performance during the COVID-19," *Procedia Computer Science*, vol. 184, pp. 835–840, Jan. 2021, https://doi.org/10.1016/j.procs.2021.03.104.

[4] P. Dabhade, R. Agarwal, K. P. Alameen, A. T. Fathima, R. Sridharan, and G. Gopakumar, "Educational data mining for predicting students' academic performance using machine learning algorithms," *Materials Today: Proceedings*, vol. 47, pp. 5260–5267, Jan. 2021, https://doi.org/10.1016/j.matpr.2021.05.646.

[5] I. Đ. Babić, "Machine learning methods in predicting the student academic motivation," *Croatian Operational Research Review*, pp. 443–461, Dec. 2017.

[6] A. Cano and J. D. Leonard, "Interpretable Multiview Early Warning System Adapted to Underrepresented Student Populations," *IEEE Transactions on Learning Technologies*, vol. 12, no. 2, pp. 198–211, Apr. 2019, https://doi.org/10.1109/TLT.2019.2911079.

[7] M. Tsiakmaki, G. Kostopoulos, S. Kotsiantis, and O. Ragos, "Fuzzy-based Active Learning for Predicting Student Academic Performance," in *Proceedings of the 6th International Conference on Engineering & MIS 2020*, New York, NY, USA, Jun. 2020, Art. no. 87, https://doi.org/10.1145/3410352.3410823.

[8] F. D. Pereira *et al.*, "Explaining Individual and Collective Programming Students' Behavior by Interpreting a Black-Box Predictive Model," *IEEE Access*, vol. 9, pp. 117097–117119, 2021, https://doi.org/10.1109/ACCESS.2021.3105956.

[9] R. Alamri and B. Alharbi, "Explainable Student Performance Prediction Models: A Systematic Review," *IEEE Access*, vol. 9, pp. 33132–33143, 2021, https://doi.org/10.1109/ACCESS.2021.3061368.

[10] W. Xing, R. Guo, E. Petakovic, and S. Goggins, "Participation-based student final performance prediction model through interpretable Genetic Programming: Integrating learning analytics, educational data

mining and theory," *Computers in Human Behavior*, vol. 47, pp. 168–181, Jun. 2015, https://doi.org/10.1016/j.chb.2014.09.034.

[11] J. Bravo-Agapito, S. J. Romero, and S. Pamplona, "Early prediction of undergraduate Student's academic performance in completely online learning: A five-year study," *Computers in Human Behavior*, vol. 115, Feb. 2021, Art. no. 106595, https://doi.org/10.1016/j.chb.2020.106595.

[12] W. Zhang, Y. Zhou, and B. Yi, "An Interpretable Online Learner's Performance Prediction Model Based on Learning Analytics," in *Proceedings of the 11th International Conference on Education Technology and Computers*, New York, NY, USA, Jan. 2020, pp. 148–154, https://doi.org/10.1145/3369255.3369277.

[13] J. Xu, K. H. Moon, and M. van der Schaar, "A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 5, pp. 742–753, Dec. 2017, https://doi.org/10.1109/JSTSP.2017.2692560.

[14] A. Cano, A. Zafra, and S. Ventura, "An interpretable classification rule mining algorithm," *Information Sciences*, vol. 240, pp. 1–20, Aug. 2013, https://doi.org/10.1016/j.ins.2013.03.038.

[15] J. Gu and V. Tresp, "Semantics for Global and Local Interpretation of Deep Neural Networks." arXiv, Oct. 20, 2019, https://doi.org/10.48550/arXiv.1910.09085.

[16] E. Angelino, N. Larus-Stone, D. Alabi, M. Seltzer, and C. Rudin, "Learning Certifiably Optimal Rule Lists," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, May 2017, pp. 35–44, https://doi.org/10.1145/3097983.3098047.

[17] L. Calvet Liñán and Á. A. Juan Pérez, "Educational Data Mining and Learning Analytics: differences, similarities, and time evolution," *RUSC. Universities and Knowledge Society Journal*, vol. 12, no. 3, Jul. 2015, Art. no. 98, https://doi.org/10.7238/rusc.v12i3.2515.

[18] M. Langer *et al.*, "What do we want from Explainable Artificial Intelligence (XAI)? – A stakeholder perspective on XAI and a conceptual model guiding interdisciplinary XAI research," *Artificial Intelligence*, vol. 296, Jul. 2021, Art. no. 103473, https://doi.org/10.1016/j.artint.2021.103473.

[19] C. B. Azodi, J. Tang, and S.-H. Shiu, "Opening the Black Box: Interpretable Machine Learning for Geneticists," *Trends in Genetics*, vol. 36, no. 6, pp. 442–455, Jun. 2020, https://doi.org/10.1016/j.tig.2020.03.005.

[20] 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 (Applications and Reviews)*, vol. 40, no. 6, pp. 601–618, Aug. 2010, https://doi.org/10.1109/TSMCC.2010.2053532.

[21] "Statement on Algorithmic Transparency and Accountability," ACM US Public Policy Office, New York, NY, USA, Jan. 2017.

[22] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, May 2019, https://doi.org/10.1038/s42256-019-0048-x.

[23] T. P. Minh, H. B. Duc, and V. D. Quoc, "Analysis of Leakage Inductances in Shunt Reactors: Application to High Voltage Transmission Lines," *Engineering, Technology & Applied Science Research*, vol. 12, no. 3, pp. 8488–8491, Jun. 2022, https://doi.org/10.48084/etasr.4826.

[24] N. L. Tran and T. H. Nguyen, "Reliability Assessment of Steel Plane Frame's Buckling Strength Considering Semi-rigid Connections," *Engineering, Technology & Applied Science Research*, vol. 10, no. 1, pp. 5099–5103, Feb. 2020, https://doi.org/10.48084/etasr.3231.

[25] H. Basarudin *et al.*, "Evaluation of Climate Change Effects on Rain Rate Distribution in Malaysia using Hydro-Estimator for 5G and Microwave Links," *Engineering, Technology & Applied Science Research*, vol. 13, no. 4, pp. 11064–11069, Aug. 2023, https://doi.org/10.48084/etasr.5552.

[26] N. N. Long, N. H. Quyet, N. X. Tung, B. T. Thanh, and T. N. Hoa, "Damage Identification of Suspension Footbridge Structures using New Hunting-based Algorithms," *Engineering, Technology & Applied Science Research*, vol. 13, no. 4, pp. 11085–11090, Aug. 2023, https://doi.org/10.48084/etasr.5983.