Dania Alsubihat, et. al. International Journal of Engineering Research and Applications
www.ijera.com
ISSN: 2248-9622, Vol. 13, Issue 4, April 2023, pp. 206-218

RESEARCH ARTICLE OPEN ACCESS

Predicting Student's Performance Using Combined Heterogeneous Classification Models

Dania Alsubihat*, Dr. Najah Al-shanableh **
**(Department of Computer Science, Al al-Bayt University, Mafraq, Jordan)*
*** (Department of Computer Science, Al al-Bayt University, Mafraq, Jordan)*

ABSTRACT
With the development of information technology, universities have become more concerned about their student's data. Therefore, educational data mining has contributed to extracting useful information from this data by analyzing and predicting student performance. This paper compares and analyses a number of the most recent algorithms, including logistic regression, K-nearest neighbour, decision tree, support vector machine, naive Bayes, multilayer perceptron, random forest, gradient boosting, Extreme Gradient Boosting (XGBoost), Categorical Boosting (CatBoost), and light gradient boosting machine (LightGBM), to predict students' academic performance. According to the analysis of the results, each of the classifiers used in the experiments produced an accurate result. However, the CatBoost algorithm produced the most accurate result compared to all others, reaching 93.15% in the student status prediction model; the XGBoost had an accuracy rate of 93%; and the RF provided a 92.9% accuracy rate. The heterogeneous model result had 93.46% accuracy.
Keywords-Extreme Gradient Boosting, Categorical Boosting, Classification,Data Mining, Heterogeneous model

Date of Submission: 07-04-2023 Date of acceptance: 21-04-2023

I. INTRODUCTION

It is essential to Digitize education because of the development of information technology and its emergence across all industries, particularly in the wake of the COVID-19 pandemic. This prompted universities and schools to pay more attention to e-learning systems, so educational materials, student data, grades, and many other data points have been entered into the educational institution systems. The increase in the number of educational institutions and students has led to an increase in student data. Consequently, there was a need for a system that analyzes the data generated by the learning environment; this system is called educational data mining (EDM). EDM uses data mining algorithms for data education [1]. EDM has an essential role in developing the educational environment by extracting and using educational data through machine learning techniques and data mining methods [2]. A major goal of EDM is to improve education and Appling data mining techniques for analyzing the data collected from students because applications of EDM are used to predict student performance [3].

The timely prediction of student performance is a strategy for achieving this goal. Predicting students' performance plays a vital role in the educational process by increasing teaching effectiveness. Therfore, universities and schools can develop plans to improve education based on student performance [5]. Predicting students' performance can identify students who are at risk, prevent students from dropping out, and help teachers decide which course materials to use [6].

Data mining has been used in many fields, including medicine, economics, advertising, and enterprise administration [7]. Discovering educational data can be beneficial for identifying information and thus acquiring an understanding that informs choices and improves the educational system. Discovering student information can be beneficial for understanding the reasons behind college students' poor overall performance and identifying the causes of certain behaviors [7]. According to the information gathered by [8], supervised learning was the most frequently employed method to predict students' conduct because it offers reliable, accurate results. In recent years, there was [4] an apparent expansion in the range of methods used to predict student achievement.

II. Educational Data Mining

I. Educational Data Mining (EDM) refers to research projects that mine datasets to address educational research objectives, such as by examining how people learn and provide instruction. These data can also come from a variety of educational contexts, including education and data management systems, interactive learning environments, intelligent tutoring systems, educational video games, and record-breaking learning sports. Data from sources such as log files, student-produced artifacts, discourse, learning content and context, sensor data, multiple-useful resources, and multimodal streams are all part of educational data mining. By enhancing data-driven understandings of learning and teaching systems in a variety of situations, educational data mining studies seek to assist students and teachers [9]. EDM applies data mining algorithms to the education domain. EDM refers to the processes designed for the analysis of data from learning environments to understand students [1].

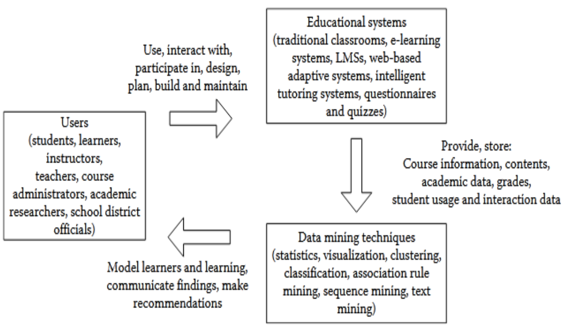

Fig. 1. Applying data mining to the design of educational systems [1]

Figure 1 demonstrates how the educational system's use of data mining is a loop in which students acquire knowledge. Applying data mining to get information from the system (courses, grades, interaction) will provide helpful information that can improve the educational system.

2.1Students' performance

The performance of students is one of the most important and difficult challenges for educational institutions. In university-level e-learning contexts, this challenge might be quite helpful [8]. Student performance is [10] integral to any educational environment. It measures the effectiveness of education and analyzes student performance to improve education quality[10].

Because of the importance of students' performance and need to measure it as an indicator of the effectiveness of education and the level of education at a university, this, in turn, improves the level of the university. In this study, we sought to measure students' performance at Al-Bayt University by using classification algorithms and heterogeneous ensemble learning techniques. Improving education is important because it gives the university, instructor, and student the opportunity to avoid any problems and improve their plans and performance.

1. Will a heterogeneous ensemble increase the predictive power compared to other models?

2. Does the use of the chosen classification algorithms provide more accuracy?

The main driving force behind the work in this study is the use of heterogeneous ensemble learning techniques and classification algorithms to improve student performance prediction. Data collection and feature selection are the first steps in the many stages of data mining. Along with the preparation, a data cleaning method is then used. The data will then be classified by using a heterogeneous methodology. The following are the study's major objectives:

- Find a new heterogeneous classification model for predicting student performance based on real data from Al al-Bayt University (ABBU) university.

- Help students to understand their performance, take action, and avoid risks.

- Evaluate and compare the accuracy of the suggested models' performance.

III. MACHINE LEARNING METHOD

Classifying methods into single methods, ensemble methods, and hybrid methods is one of the most well-known classification strategies. Machine learning (ML) methods are continuously progressing to provide better overall performance algorithms. Hybrid and ensemble techniques often perform better than single ML methods [11].

To develop machine precise, reliable learning methods, ensemble and hybrid methodologies have both been proposed [11]. To improve the methodology, hybrid machine learning methods have been created by combining one ML method with another ML method or by alternative computing and optimization techniques. The future success of ML will depend on the development of ensemble and hybrid techniques, which are produced using different gathering strategies such as bagging or boosting to have more than one ML classifier [11].

ML approaches are being used increasingly frequently, particularly in hybrid and ensemble methods [11].

3.1 Ensemble Method

Ensemble approaches combine single models to increase the model's predictability and stability, providing a greater level of predictive performance than a single model. To reduce variance through bagging and bias through boosting or increase prediction using stacking, ensemble learning looks for ways to combine different machine learning models into one predictive model [12].

Additionally, rather than only using one ML classification tree, ensemble approaches may employ several trees; this technique improves the accuracy of the model [11]. To increase training accuracy and, by extension, testing accuracy, ensemble techniques, which are supervised learning algorithms, use special training algorithms [11]. Ensemble methods also enable training algorithms to do soft training; ensemble approaches include bagging, boosting, and random forest. They have been employed in various sectors and can be used in research involving a significant amount of generic data. Bagging, stacking, and boosting are the three standard ensemble learning techniques.

3.1.1 Bagging

According to [14], a way to build an ensemble learning algorithm, where each base model trains on a different set of data, is to use a bootstrap aggregation (bagging), invented by Breiman. Figure (2) shows how bagging techniques produce one strong model.

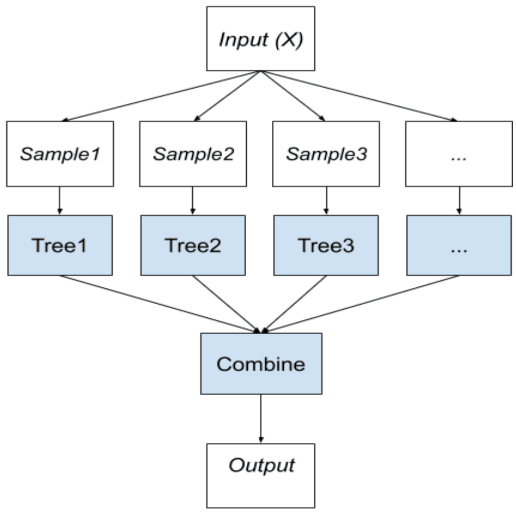

Fig. 2. Bagging techniques

- Voting

According to [15], one of the most logical, simple ensemble types is this one. The base classifiers for the voting classifier may be of the same type or a different one, making it a homogeneous and heterogeneous version of ensemble learning. As previously noted, this ensemble functions as an extension of bagging (e.g., random forest).

A voting classifier's architecture consists of "n" machine learning models, each of which has predictions that are evaluated in one of two ways: hard or soft. In the hard mode, the forecast with "the most votes" is the correct one. The winning class will have the highest weighted and averaged probability. However, the voting classifier in soft mode considers the probabilities generated by each ML mode.

- Random Forest

A random forest is [16],[17] one of the ensemble techniques. In simple terms, it is a bagging technique or bootstrap aggregation that combines multiple base learner models called decision trees (DT). As shown in Figure 3, a dataset is divided into subsets according to its features, and each subset of data will be represented in the DT. The output from each tree will be combined into a final output through the majority voting.

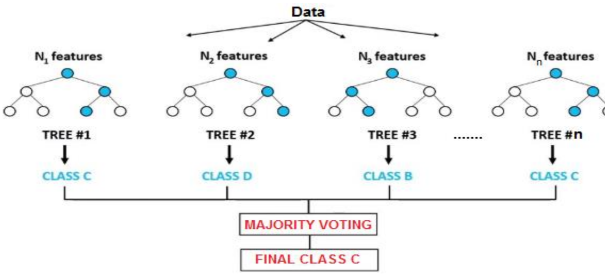

Fig. 3. Random forest

Using multiple trees will reduce overfitting because one tree has low bias and high variance. Low bias means that creating a DT in depth will probably get trained for the training set, which significantly reduces training errors.

3.1.2 Stacking

Stacked generalization (stacking) is an ensemble technique that uses different model types based on the training data and a model to combine predictions to identify a diverse group of members [12]. The term "level-1 model" refers to a model that is used to combine the predictions in stacking, and "level-0 models" refer to the ensemble members of the ensemble. Although there might be more layers of models, the most popular hierarchy has two levels. For instance, we might have three or five level-1 models instead of just one, and a single level-2 model combines the level-1 models' projections to produce a prediction [18]. Figure 4 shows how stacking techniques create one strong model.

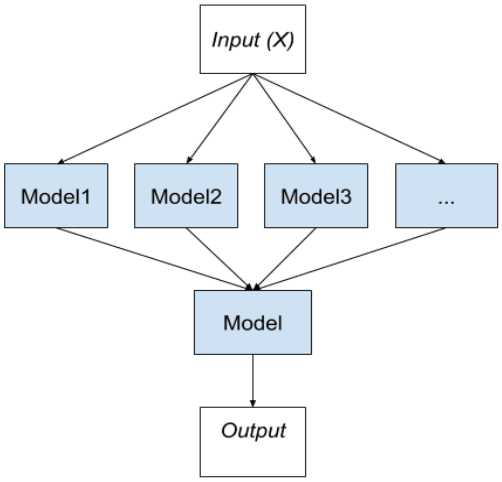

Fig. 4. Stacking techniques

3.1.2 Boosting

Boosting is an ensemble technique called aims that alters the training data to draw attention to instances in which past-fit models for the training dataset have misfit [12]. The concept of correcting prediction errors is the fundamental characteristic of boosting ensembles. Figure 5 shows how boosting techniques are done to get one strong model.

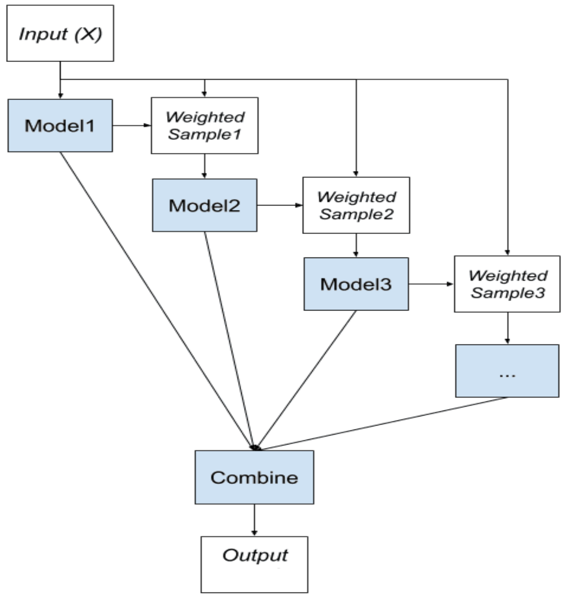

Fig. 5. Boosting technique

To attempt to correct the predictions of the first model, the models are fitted and introduced to the ensemble one at a time. The third model then attempts to correct the predictions of the second model, and so on. The learning algorithm is typically altered to pay more or less attention to examples (rows of data) depending on whether they have been properly or erroneously predicted by previously added ensemble members while leaving the training dataset largely untouched. For instance, the rows of data can be weighted to show how much attention a learning algorithm needs to provide while developing the model [15]. Many popular ensemble algorithms are based on this approach, including AdaBoost (canonical boosting), gradient boosting machines, and stochastic gradient boosting (XGBoost and similar ones).

- *Gradient Boosting*

Boosting algorithms iteratively convert poor learners who are only marginally better than random into strong learners [19]. A regression approach like boosting is called gradient boosting. Boosting gradients is regarded as an ensemble technique [20]. This algorithm sequentially combines various predictors with some shrinkage on them [21]. Gradient bosting (GB) is a powerful ML technique that produced the best outcomes in a variety of real-world tasks. Over the years, it has become the go-to technique for resolving learning issues with a variety of characteristics, noisy data, and complex dependencies [22]. Several efficient techniques that focus on both computation speed and accuracies, such as LightGBM, XGBoost, and CatBoost, have recently been added to the family gradient-boosting algorithms [19].

1. XGBoost

Extreme Gradient Boosting (XGBoost) is a DT ensemble based on gradient boosting that was created to be scalable [25]. XGBoost constructs an additive expansion of the objective function by minimizing a loss function, like gradient boosting. A different loss function is used to regulate the complexity of the trees because XGBoost uses only DTs as its basic classifiers [19]. Shrinkage, a further regularization hyper-parameter in XGBoost, lowers the step size in additive expansion. Finally, other methods, such as the depth of the trees, can be used to limit the complexity of the trees. The models may be trained more quickly and with less storage space thanks to the reduction in tree complexity. To decrease overfitting and speed up training, XGBoost also uses randomization approaches. Random subsamples are used to train individual trees, and column subsampling is used at the tree and tree node levels as a randomization strategy in XGBoost. Additionally, by designing a function that outputs the gradient and the hessian (second-order gradient) and running it via the “objective” hyperparameter, XGBoost may be expanded to accommodate any user-defined loss function [19].

2. LightGBM

Gradient boosting is implemented in LightGBM [27], a sizable library that also suggests some variations. Gradient boosting has been implemented in this library with a special emphasis on developing a computationally efficient approach based on the precomputation of the feature histogram, like in XGBoost. Tens of learning hyperparameters are also available in the library, enabling this model to be used in many contexts: The implementation is CPU and GPU compatible, capable of doing simple gradient boosting, and has a variety of randomizations (column randomization, bootstrap subsampling).

3. CatBoost

A gradient-boosting library called CatBoost [28] tries to reduce the prediction shift that happens during training. A series of base models that do not include that occurrence in their training set is used to estimate the gradients, as suggested in CatBoost [28]. CatBoost initially adds a random permutation to the training cases to achieve this. Building i = 1, . . . , N base models for each of the M boosting iterations is the idea behind CatBoost (but not the actual implementation). The gradient of the i+1 instance for the (m + 1) th boosting iteration is estimated using the ith model of the mth iteration, which is trained on the first i instances of the permutation. This process is repeated using s different random permutations to become independent of the starting random permutation. Despite this, the CatBoost implementation is optimized, so only one model, capable of handling all permutations and models, is created per iteration. Symmetric trees serve as the basis models (or decision tables). By using the same split condition to extend all leaf nodes level wise, these trees are produced.

3.2 Hybrid Method

To perform better and produce the best results, hybrid methods combine multiple ML classifiers or soft computing techniques. Hybrid methods may include one unit for prediction and one unit for the optimization of the prediction unit to provide an accurate result [11]. Hybrid techniques profit from the gains of two or more methods to perform better.

Therefore, it may be argued hybrid techniques combine many single ML methods to create a new method that is more flexible and robust than a single ML method [11]. Hybrid methods have become well known because of their high potential and capacity. Hybrid methods are like companies with exclusive employees with good experience and skills who help reach its only target. Hybrid techniques combine machine learning models to enhance the performance of a single model, but the hybrid model could have an optimization model for other prediction models to provide greater accuracy [29]. According to [30], using a hybrid algorithm has positives when the dataset is complex and does not give the desired accuracy when classifying it by one classifier. The hybridization trend in machine learning is unlimited. Every single ML can make a new hybrid model in many ways, such as through architectural integration, data manipulation, and model parameter optimization [31].

3.2.1 Ensemble vs. Hybrid Model Machine Learning

Combing classifiers to enhance performance is a new trend in ML [32]. Hybrid and ensemble learning has significant importance to scholars. Ensemble models and the hybrid method both use the integration principle but with a simple difference because ensemble ML combines a homogenous model unlike a hybrid classifier, which merges heterogeneous models [32]. The ensemble classifier is based on models that are grouped together to make a group decision for prediction. A hybrid classifier considers more filtering features; it is called "hybrid" because it is interested in data preprocessing and model creation, and unlike an ensemble, there are no restrictions for the data processing. In hybrid ML, each model is classified by one classifier [33].

IV. RESEARCH METHODOLOGY

In this research, the heterogeneous model was used to combine more than one classifier to produce a strong predictive model to predict student performance by using real data. The heterogeneous model combines XGBoost, CatBoost, and random forest (RF).

First, the data was collected from the ABBU student system. Then, we prepared the data (cleaning it and making feature selections) into a format that was acceptable for the classification algorithm. After that, the dataset was ready to use. Because we taught the model in this step, we split the data into a 70% training set and a 30% testing set. The training set was trained by using a heterogeneous ensemble ML strategy, in which we anticipated the outcome of the three combined algorithms: CatBoost, RF, and XGBoost using soft majority voting (XGBoost) (We selected these algorithms based on the literature research because they were the most accurate and efficient in most studies and demonstrated the best accuracy in the experimental stage of this research). In the heterogeneous ensemble classifier, we modified the quantity of input features without using the programming procedure (selecting a subset of the features rather than all the features). After having an output from the three models, we applied the majority vote to create the final output. In the end, at the validation step, our model's accuracy, precision, and recall was measured. Evaluations included confusion matrix calculations, comparing our model to other models, and a final determination of whether the model met the established business goals. This study aims to predict the academic status of students and examine the effectiveness of various predictive models on their academic performance. Six phases comprised the suggested methodology: data collection, data preprocessing, feature selection, model development and training, model testing, and evaluation outcomes. A summary of the heterogeneous classification model's methodology is shown in Figure 6 for predicting student performance.

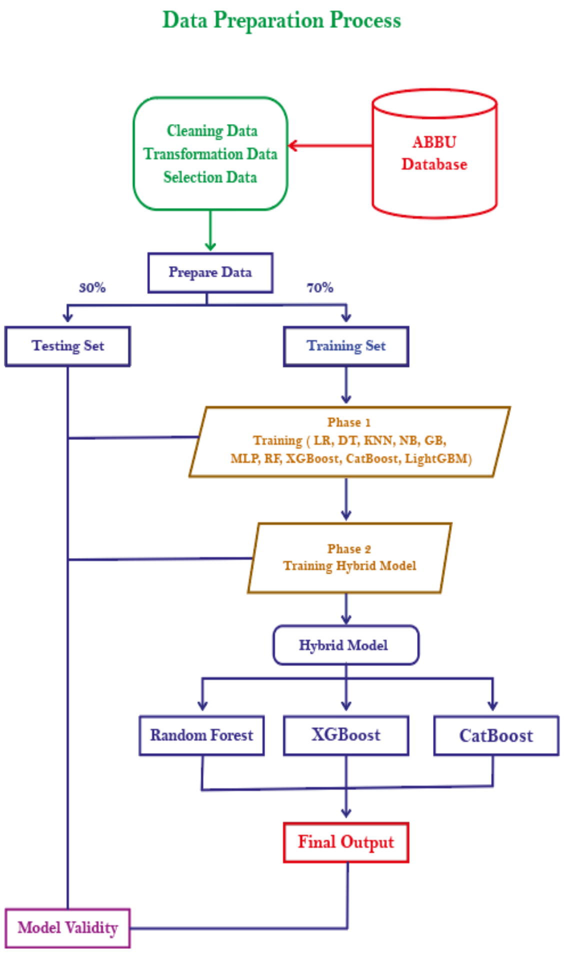

Fig. 6. Methodology flow chart

4.1 Feature (Data) Selection

The target variable was used in supervised feature selection approaches, such as strategies that eliminate irrelevant variables. Selecting characteristics can be further subdivided into wrapper and filter methods, which is another approach for thinking about the mechanism being employed. These techniques are usually supervised, and the effectiveness of the resulting model on a hold-out dataset serves as the basis for evaluation [37].

Instead of focusing on cross-validation performance, filter approaches capture the inherent characteristics of the features assessed by univariate statistics. Compared to wrapper methods, these techniques are quick and less computationally intensive [38]. Models are created used different subsets of the input features via wrapper-feature selection methods, and the features that provide the model with the highest performance score based on the performance measure are then selected [37].

- Information Gain

Information gain is used to determine how much entropy will be lost as a result of a dataset update. It can be used for feature selection by evaluating the information gain of each variable in relation to the target variable [37].

Figure (7) shows the feature importance using the information gain method for the prediction statues model feature.

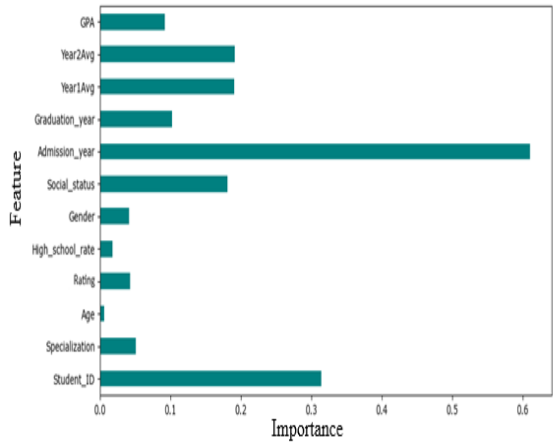

Fig. 7. Feature selection based on importance for student status prediction

As shown, the admission year feature was ranked first in importance, and it is logical to predict students' state, the less important are high school rate and age.

- Recursive Feature Elimination

Given an external estimator that assigns feature weights, recursive feature elimination (RFE) is a technique for selecting features by continually taking into account smaller and smaller sets of features (for example, the coefficients of a linear model). Using the coef_ attribute or the feature importances_ attribute, the estimator is first trained on the original set of features to identify the importance of each one. The remaining features are then reduced in importance, starting with the least important ones. Once the appropriate number of features for selection has been reached, the technique is recursively repeated on the pruned set [37]. The box and whisker plot constructed for the distribution of accuracy scores for each specified number of features is shown in Figure 8.

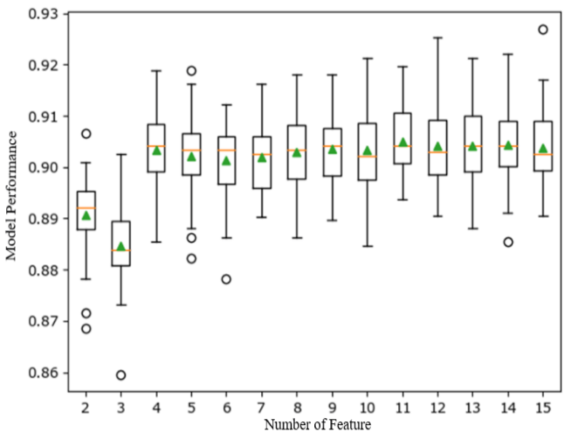

Fig. 8. Box plot of RFE number of selected features

Because only 12 features are relevant to the target variable, we can see that performance improves as the number of features increases and may peak between 12 and 14, as was anticipated. Last but not least, Table 1 displays the following features from the dataset.

TABLE 1
FEATURE DESCRIPTIONS

Feature’s Name	Feature Description
Student_ID	A unique numerical variable for each student
Specialization	Includes 42 majors (encoded from 1 to 42)
Study_status	Includes six basic statuses: Graduate: 1, Regular: 2, Expected to Graduate: 3, Dismissed: 4, Cut Off: 5, Withdraw from University: 6.
Age	A numerical variable is the age of the student accepted into the university
High_school_rate	A numerical variable; the Tawjehe-Rate of each student
Gender	Student’s gender numerical variable; 1 for male and 2 for female
Social_status	A numerical variable; Single: 0, Married: 1, Divorced: 2
Admission_year	A numerical variable representing the year of acceptance into the university
Graduation_year	A numerical variable representing the year of finishing the study
Year1Avg	A numerical variable representing the average of student grades at the first year
Year2Avg	A numerical variable representing the average student grade in the second year
GPA	A numerical variable is the average student grade (CGPA)
Rating	A string variable symbolizing the GPA in ABBU instruction: Excellence ≥ 94, Excellent ≥ 84, Very good ≥ 76, Good ≥ 68, Acceptable ≥ 60, Weak ≥ 50, Fail < 50.

V. RESULT AND DISCUSSION

Understanding and using descriptive statistics with the aid of visualization and plots requires a critical analysis of the student’s academic characteristics [39]. By providing the information in a visual context through charts and graphs, data visualization sits squarely at the heart of our analysis, giving us a clear understanding of what the data means. Further, it has resulted in the provision of a usable method for data simplification and formatting and comprehension of outliers and patterns in the data while improving decision-making when using data [40]. The dataset contains 19,700 records for students, divided into 10,667 female and 9,034 male students. For the Social Status feature, students were visualized into three categories based on their social status: there were 18,995 single students, followed by 649 married students, and the divorced category of 56 students.

In the feature called Rating, we specify the rate of each student based on the marking and grading method of BSc students for AABU. Figure 9 illustrates the seven categories into which the student rating in the Rating feature was divided.

• Excellence: This is the recognition of students with average GPAs of 94 or higher; in our sample, 43 students received an Excellence rating.

• Excellent: Students with a GPA ranging from 84 to 93.99%, and 1,609 students in our dataset received an Excellent rating.

• Very Good: 4,597 students in our dataset received this grade, which is given to students with a GPA ranging from 76 to 83.99%.

• Good: 5,838 students received this commendation, making it the most common one awarded to students with average GPAs between 68 and 75.99%.

• Acceptable: According to our dataset, 4,443 students received this grade, which is given to students with an average GPA between 60 and 67.99%.

• Weak: According to our dataset, 2,440 students received this grade, which is given to students with an average GPA of 50 to 59.99%.

• Failed: In our dataset, 731 students who had an average GPA of 50 or less were considered to have failed.

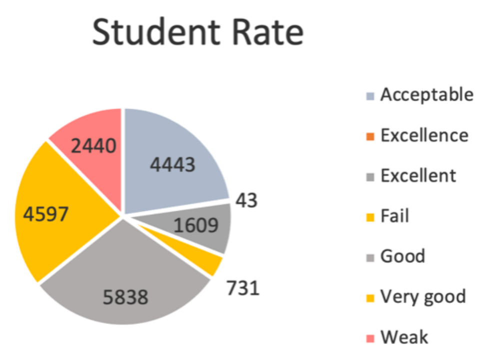

Fig. 9. Visualization of student rate

In addition, the student's data was collected from many university colleges and majors. The basic two types of university majors are humanistic/literary majors and science majors. The most popular discipline for study was law with 1,209 students, followed by 957 students studying Grade teacher major.

The student status was examined as the target class's most crucial characteristic, and the status characteristic of Al-al-Bayt university students had numerous categories, including six fundamentals:

1. Graduate: Students who graduated from the university and moved on are considered to be in this category. There were 2,598 students in this category.

2. Regular: This refers to if the students are still enrolled in school and up until they graduate; 13,166 students were given this rank.

3. Expected to Graduate: There are 82 students in this status, and it includes students who need only a few more hours to complete their specialization hours.

4. Cut Off: In this situation, students leave the institution without formally withdrawing; this also includes students enrolled in special studies who have the chance to improve their academic standing for one or three semesters; 783 students are included in this category.

5. Dismissed: This refers to a situation in which a student is expelled from a school or specialty for moral or academic reasons. This category also includes those who have been temporarily withdrawn from classes and includes 253 other students who were placed on academic suspension for the semester (disciplinary).

6. Withdraw from University: In this case, the student left the university unofficially, and 2,819 students fall into this category.

In this research, the dataset was 49,235 rows, and the prepared data included 19,700 records. The prediction model was built and tested to predict study status, which describes the student's status if they graduated, studied at, or withdrew with from the university. As a first step, we trained the model (logistic regression, DT, support vector machine, K-nearest neighbor, naïve Bayes, multi-layer perceptron, gradient boosting, random forest, XGBoost, CatBoost, and LightGBM) in each of these algorithms separately. After using the classification methods on the dataset, the results were distinguished using different evaluation criteria and computed based on the confusion matrix, and we determined their accuracy as shown in Table 2.

In the first experiment, the classification algorithms (LR, DT, SVM, K-NN, NB, XGBoost, MLP, RF, CatBoost, XGBoost, LightGBM) were simply carried out one at a time on the dataset of 19,700 records with 12 features as inputs, one as an output, and six state instances representing the status of the student: Graduate, Regular, Expected to Graduate, Dismissed, Cut Off, and Withdraw from University. As we previously indicated, a confusion matrix was used to create the four performance measures (Accuracy, Precision, Recall, and F-Score) that were employed in the research to assess classification outcomes.

TABLE 2
MODEL EVOLUTION FOR PREDICTING STUDENT STATUS

Algorithm	Precision	Recall	F1-Score	Accuracy
LR	0.43	0.66	0.52	0.67266
DT	0.86	0.90	0.88	0.89477
SVM	0.57	0.72	0.63	0.71588
KNN	0.87	0.89	0.88	0.88223
NB	0.59	0.73	0.65	0.72349
GB	0.92	0.92	0.92	0.92451
MLP	0.43	0.66	0.52	0.67266
RF	0.90	0.92	0.91	0.92922
CatBoost	0.93	0.93	0.93	0.93154
XGBoost	0.92	0.93	0.92	0.93002
LightGBM	0.91	0.92	0.92	0.92871

Based on these two results, CatBoost had the highest accuracy for making predictions (students' status), followed by RF and XGBoost; however, other algorithms provided similar results but were less accurate compared to the above algorithms.Based on these results, the heterogeneous model was created by merging CatBoost, RF, and XGBoost (repeated 10 times to give more power to this model). The accuracy rate for this heterogeneous model, which was applied to a dataset of 19,700 records with 13 features as inputs and one output, was 93.46%. The precision, recall, and F1-score rates for this model's predictions of the students statuses were 91%, 92%, and 91%, respectively, according to Table 3. The ROC curve graph for this heterogeneous model's prediction of student status is shown in Figure 10.

TABLE 3
HETEROGENOUS MODEL EVOLUTION

Evaluation Measure	Precision	Recall	F1-Score	Accuracy
Heterogeneous Model	0.91	0.92	0.91	0.9346

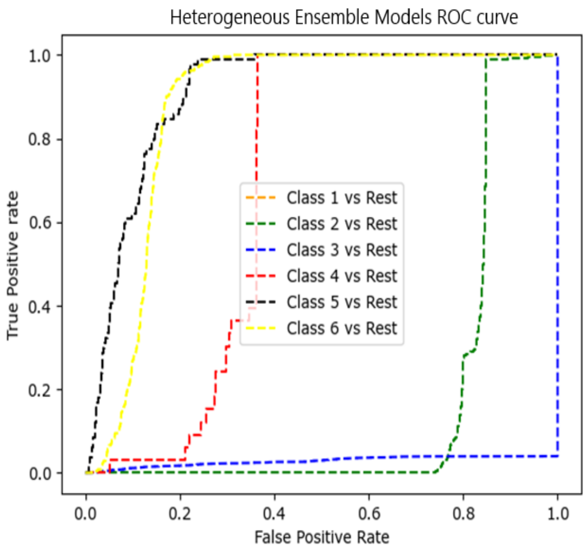

Figure (10): ROC Curve for the Heterogeneous Model for Predicting Student Status

The same dataset from AABU in Jordan was used in two studies [35] and [36]. Table 4 shows the results of these studies. The first study [35] suggested a prediction model based on the XGBoost classifier using the same dataset, and it produced the best accuracy reach of 77%, demonstrating that our proposed model produced better results considering the classification accuracy reach of 93%. Bases on these results, the DT accuracy was 76%, KNN accuracy was 75.71%, and SVM accuracy was 75.69%. The results for these algorithms in straight order were 89.47%, 88.22%, and 71.55%, which verifies the usefulness of our proposed sophisticated preprocessing approaches and potent contemporary algorithms.

The same dataset from the first study was used in the second study [36] because she created an XGBoost classifier with an accuracy reach of 91.61% and a LightGBM classifier with a 91.95% accuracy reach. This demonstrates our suggested approach produced superior results given the classification accuracy reaches of 93% and 92.8%. This also demonstrates the value of our suggested sophisticated preparation approaches and application of potent contemporary algorithms. However, we examined other gradient boosting techniques she applied. The final status prediction model produced by the CatBoost algorithm had a prediction accuracy of 92.16%, according to the performance output findings. But it reached 93.15% in the model we have created.

TABLE 4
PREVIOUS STUDIES' RESULTS

Reference	Proposed Algorithms	Dataset	Accuracy Results
[36]	XGBoost, LightGBM, and CatBoost	From Al-al-Bayt University in Jordan	XGBoost: 91.61% LightGBM: 91.95% CatBoost: 92.16%
[35]	XGBoost, K-Nearest Neighbor (KNN), RF, Support Vector Machine (SVM), and DT	From Al-al-Bayt University in Jordan	SVM: 75.69% KNN: 75.71% DT: 76% RF: 76.86% XGBoost: 77%

In general, the proposed heterogeneous model's accuracy reached 93.46% for predicting the final student's status model. Finally, we achieved our goal of evaluating and comparing the accuracy of the proposed models.

VI. CONCLUSION

The use of technology in education has produced enormous volumes of data, which is the focus of our work. We used real data from the Al-Bayt University student dataset, analyzed and preprocessed the dataset, and identified most of the characteristics or factors that influenced students' performances in the university and student performance prediction models. Therefore, the data was cleaned and processed to train various data mining models to define the classification method.

We selected the heterogeneous models based on the best algorithms from the available options. We revealed that CatBoost performed substantially better than all other classifiers, with accuracy for the predicting the student's status model reaching 93.15%. That was followed by XGBoost classification, achieving an accuracy reach of 93%. Finally, at the end, the RF algorithm prediction model accuracy reached 92.92%. Compared to the other algorithms listed above, these three algorithms provided the best accuracy. The heterogeneous proposed model (which combines XGBoost, CatBoost, and RF) provides an accuracy reaching 93.46% for predicting the final Student's Status model. A model based on a heterogeneous ensemble classification technique was developed to predict students' academic performance, and when compared to other models used for the same dataset, the performance results demonstrated better accuracy. We also found the optimum subset of features that the input data can represent effectively and significantly for affecting the prediction results.

REFERENCES

[1]. C. Romero, S. Ventura, M. Pechenizkiy, and R. S. Baker, Handbook of Educational Data Mining, Chapman & Hall/CRC Data Mining and Knowledge Discovery Series, 1st ed. CRC Press, 2010.

[2]. B. Albreiki, N. Zaki, and H. Alashwal, "A systematic literature review of students' performance prediction using machine learning techniques," Education Sciences, vol. 11, no. 9, p. 552, 2021, https://doi.org/10.3390/educsci11090552.

[3]. H. L. Alamri, S. R. Almuslim, M. S. Alotaibi, D. K. Alkadi, I. U. Khan, and N. Aslam, "Predicting student academic performance using support vector machine and random forest," presented at ICETM 2020, London, United Kingdom. https://doi.org/10.1145/3446590.3446607.

[4]. V. Nedeva and T. Pehlivanova, "Students' performance analyses using machine learning algorithms in WEKA IOP Conf. Ser," Mater. Sci. Eng 1031 012061.

[5]. A. B. Zorić, "Predicting students' academic performance based on enrolment data," International Journal of Innovation and Economic Development, 2020, http://dx.doi.org/10.18775/ijied.1849-7551-7020.2015.64.2004.

[6]. L. M. Abu Zohair, "Prediction of student performance by modelling small dataset size," International Journal of Educational Technology in Higher Education, vol. 16, no.1, 2019, https://doi.org/10.1186/s41239-019-0160-3.

[7]. A. Samah, "Students' performance evaluation using machine learning algorithms," College of Basic Education Research Journal, Dec. 2020, doi:10.33899/berj.2020.166006.

[8]. J. L. Rastrollo-Guerrero, J. A. Gómez-Pulido, and A. Durán-Domínguez, "Analyzing and predicting students' performance by means of machine learning: a review," Applied Sciences, vol. 10, no. 3, p. 1042, Feb. 2020, https://doi.org/10.3390/app10031042.

[9]. C. Baek and T. Doleck, "Educational data mining versus learning analytics: a review of publications from 2015 to 2019," Interactive Learning Environments, 2021, doi:10.1080/10494820.2021.1943689.

[10]. A. M. Shahiria, W. Husaina, and N. Abdul Rashida, "The third information systems international conference: a review on predicting student's performance using data mining techniques," Procedia Computer Science, vol. 72, pp. 414–422, 2015, www.sciencedirect.com.

[11]. A. Mosavi, S. Ardabili, Várkonyi-Kóczy, R. Annamária, "Lecture notes in networks and systems, engineering for sustainable future, volume 101," in Selected Papers of the 18th International Conference on Global Research and Education Inter-Academia, 2019, 10.1007/978-3-030-36841-8(), – . doi:10.1007/978-3-030-36841-8.

[12]. C. Zhang, and Y. Ma, Ensemble Machine Learning: Methods and Application, 2012 ed. Springer, 2012.

[13]. O. González-Recio, J. Jiménez-Montero, and R. Alenda, "The gradient boosting algorithm and random boosting for genome-assisted evaluation in large data sets," Journal of Dairy Science, vol. 96, no. 1, pp. 614–624, 2013, https://doi.org/10.3168/jds.2012-5630.

[14]. L. Breiman, "Bagging predictors," Machine Learning, vol. 24, no. 2, pp.123–140, 1996, https://doi.org/10.1007/bf00058655

[15]. Z. Zhou, Ensemble Methods: Foundations and Algorithms, Chapman & Hall/CRC Machine Learning and Pattern Recognition, 1st ed. Chapman and Hall/CRC, 2012.

[16]. L. Breiman, "Random forests," Statistics Department, University of California, Berkeley, 2001.

[17]. G.´ Biau, "Analysis of a random forests model," Journal of Machine Learning Research, vol. 13, pp. 1063–1095, 2012.

[18]. L. Rokach, Pattern Classification Using Ensemble Methods (Machine Perception and Artificial Intelligence), World Scientific Publishing Company, 2009.

[19]. C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, "A comparative analysis of gradient boosting algorithms," Artificial Intelligence Review, vol. 54, no. 3, pp. 1937–1967, 2020, https://doi.org/10.1007/s10462-020-09896-5.

[20]. T. Hastie, R. Tibshirani and J.H. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, New York, NY: Springer, 2009.

[21]. J.H. Friedman, "Greedy functions approximation: A gradient boosting machine," Ann. Stat., vol. 29 pp. 1189–x, 2011.

[22]. G. Huang, L. Wu, X. Ma, W. Zhang, J. Fan, X. Yu,W. Zeng, and H. Zhou, "Evaluation of CatBoost method for prediction of reference evapotranspiration in humid regions," J. Hydrol, vol. 574, pp.1029–1041, https://doi.org/10.1016/j.jhydrol.2019.04.085.

[23]. S. Roy, and A. Garg, "Predicting the academic performance of students using classification techniques," in 2017 4th IEEE Uttar Pradesh Section International Conference on Electrical, Computer, and Electronics, presented at the 2017 4th IEEE Uttar Pradesh Section International Conference on Electrical, Computer, and Electronics, IEEE, Mathura, pp. 568–572, https://doi.org/10.1109/UPCON.2017.8251112

[24]. E. A. Daoud, "Comparison between XGBoost, LightGBM, and CatBoost using a

home credit dataset," 2019, https://doi.org/10.5281/ZENODO.3607805.

[25]. T. Chen, and C. Guestrin, "Xgboost: a scalable tree boosting system," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, 2016, pp. 785–794.

[26]. L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, Classification and Regression Trees, New York, NY: Chapman & Hall, 1984.

[27]. G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T. Y. Liu, "Lightgbm: a highly efficient gradient boosting decision tree," in I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, Vishwanathan, 2017.

[28]. L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, A. Gulin, "Catboost: unbiased boosting with categorical features," in Advances in Neural Information Processing Systems, vol. 31, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett (eds.), pp. 6638–6648.

[29]. S. Ardabili, A. Mosavi, and A. R. Várkonyi-Kóczy, "Advances in machine learning modelling reviewing hybrid and ensemble methods," 2019, doi: 10.20944/preprints201908. 0203.v1).

[30]. Lal, Abhishek, and C. R. S. Kumar, "Hybrid classifier for increasing accuracy of fitness data set," presented at the IEEE 2017 2nd International Conference for Convergence in Technology, Mumbai, April 7–9, 2017.

[31]. F. Anifowose, "Hybrid machine learning explained in nontechnical terms," JPT, Feb. 6., https://jpt.spe.org/hybrid-machine-learning-explained-nontechnical-terms

[32]. M. H. Motazm Khorshid, H. M. Tarek, Abou-El-Enien, M. A. Ghada, and Soliman, "Hybrid classification algorithms for terrorism prediction in the Middle East and North Africa," International Journal of Emerging Trends & Technology in Computer Science, 2015.

[33]. T. T. Wong, N. Y. Yang, and G. H. Chen, "Hybrid classification algorithms based on instance filtering," Information Sciences, vol. 520, pp. 445–455, 2020, https://doi.org/10.1016/j.ins.2020.02.021.

[34]. A. Ashraf, S. Anwer, and M. G. Khan, "A comparative study of predicting student's performance by the use of data mining techniques," American Scientific Research Journal for Engineering, Technology, and Sciences, vol. 44, no. 1, pp. 122–136, 2018, http://asrjetsjournal.org/

[35]. S. Al-husban, "Predicting the students' performance inJordian universities using machine learning," JUST-Jordan University of Science and Technology, 2021.

[36]. E. Mashagba, "Predicting the students' academic performance usinggradient boosting algorithms," Al-al Bayt University/Prince Hussein Bin Abdullah College for Information Technology, 2022.

[37]. J. Brownlee, "Data preparation for machine learning: data cleaning, feature selection, and data transforms in Python," Machine Learning Mastery, 2020.

[38]. M. Dash, and H. Liu, H., "Feature selection for classification," Intell. Data Anal., vol. 1, pp. 131–156, 1997, https://doi.org/10.1016/S1088-467X(97)00008-5.

[39]. D. J. Hand, "Principles of data mining," Drug Saf., vol. 30, pp. 621–622, 2007, https://doi.org/10.2165/00002018-200730070-00010

[40]. M. Mani, and S. Fei, "Effective big data visualization," in Proceedings of the 21stInternational Database Engineering & Applications Symposium on IDEAS 2017, presented at the 21st International Database Engineering & Applications Symposium, Bristol, United Kingdom, pp. 298–303,https://doi.org/10.1145/3105831.3105857