INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)

www.ijprems.com
editor@ijprems.com

Vol. 03, Issue 05, May 2023, pp : 480-484

e-ISSN : 2583-1062

Impact Factor : 5.725

STUDENT PERFORMANCE ANALYSIS USING MACHINE LEARNING

Saurav Kumar Gupta[1], Ishu Kumar[2], Manish Kumar[3], Sonam Kumari[4], Suchitra Devi A[5]

[1,2,3,4] Student, Dept of CSE, Sambhram Institute of Technology, Bangalore, Karnataka, India

[5]Asst Professor, Dept of CSE, Sambhram Institute of Technology, Bangalore, Karnataka, India

DOI: https://www.doi.org/10.58257/IJPREMS31181

ABSTRACT

Universities are facing significant challenges in analyzing their students' performance due to the vast amount of digital data available from sources like social media, research, agriculture, and medical records. Admission, student placement, and curriculum are among the most crucial challenges, with data analysis primarily taking place during admission and placement processes. A university's market position and reputation depend heavily on its students' academic performance, placement, and other factors. To better comprehend student performance and categorize them, this project is using processing techniques. While most universities have their management systems to manage student records, lecturers at UNIMAS lack access due to privacy settings. The proposed Student Performance Analysis System (SPAS) aims to resolve this issue by tracking students' results and using a predictive system to identify those likely to perform poorly in their courses. The system employs data mining techniques, primarily classification, to establish principles for predicting student performance.

Keywords: Data Analysis, Machine Learning, Classification Techniques.

1. INTRODUCTION

Machine learning is a subset of the broader field of artificial intelligence (AI). Its goal is to comprehend the complexities of many forms of acquired data and to determine the best model for the data by evaluating various models. This procedure is simplified for human perception and application. Machine learning is an engineering science topic; however, it differs from basic computing methods used for problem solving. Algorithms in machine learning are designed to allow a system or computer to process input data, construct training sets, and produce the desired range of defined output using statistical estimation. One critical component of a student's personal and professional development is performance evaluation. Performance evaluations highlight a student's strengths and areas of expertise. It acts as a valuable tool in enhancing their strengths and identifying areas that require improvement as goals. By being able to analyze the performance of their students, teachers can focus their attention on the necessary areas, advise and guide the scholars along the right path, and acknowledge and reward their achievements. Performance review is an important part of a student's personal and professional growth. Performance evaluations emphasise a student's areas of skill and abilities. It is a useful tool for boosting their strengths and identifying areas for progress as goals. Teachers can focus their attention on the relevant areas, counsel and guide scholars along the appropriate route, and praise and reward their achievements if they can analyze their pupils' performance.

2. METHODOLOGY

There are numerous key stages in the creation of the Student Performance Analysis system. Understanding the problem and the data is the first step in accomplishing the project's goals.

Problem and data understanding

In this phase, the functional issues with the system are identified, examined, and solutions are developed for each issue. Interviews are conducted with stakeholders and subject matter experts in the field of machine learning to better understand the system, its operation, and its limitations. To determine the needs and potential for the proposed system, further comparable systems are researched and examined. The results from previous semesters and the percentage of marks earned in secondary and senior secondary schools are among the student information gathered during this period. The characteristics of the dataset gathered for processing categorization are shown in Table 1. The success of the Student Performance Analysis system depends on effective problem and data understanding, which will also serve as the starting point for other stages of the project's development.

TABLE I. Attributes Of Dataset

Attributes	Definition
Gender	Categorical (0=Female, 1=Male)
Age	Numeric
Course	Computer Science CS=1, Information Technology IT=2
Year	2nd Year,3rd Year,4th Year
Socioeconomical Status	Low:1, Lower middle:2, Middle:3, Upper Middle:4, Upper but not rich:5, Rich:6
Working Student	Yes=1, No=0
Scholar	Yes=1, No=0
Personality Type	INTJ:0, ISTJ:1, ESFJ:2, ISFP:3, ISFJ:4, ENFJ:5, INFJ:6, ISTP:7, INFP:8, ESFP:9, ESTJ:10, ENFP:11, ESTP:12, ENTJ:13, INTP:14, ENTP:15
Time Management TM	Never=0, Always=1, Sometime=2
Class Attendance and Participation CAP	Never=0, Always=1, Sometime=2
General Study Strategy GSS	Never=0, Always=1, Sometime=2
Exam Preparation	Never=0, Always=1, Sometime=2
Note Taking	Never=0, Always=1, Sometime=2
Available Resources Usage	Never=0, Always=1, Sometime=2
Internet Connectivity Usage	Never=0, Always=1, Sometime=2
Internet Connectivity Status	Never=0, Always=1, Sometime=2

2.1 System analysis and design

The entire flow of the system is planned, analysed, and designed throughout the system analysis and design phase. The first stage is to analyse the system and user needs, which will be listed in table format. A Data Flow Diagram (DFG) is used to chart the system's input, processes, and output, which provides an overall view from the context diagram up to the first level.

Furthermore, the proposed system's logical architecture is built to ensure that the generated system performs as envisioned. The logical design is drawn using Entity-Relationship Diagrams (ERD). ERD graphically shows data items, properties, and relationships between tables in a database. The proposed system's architecture also includes a full description of the backend, which includes databases, and the frontend user interface.

3. MODELING AND ANALYSIS

Fluid and Material which are used is presented in this section. Table and Fluid should be in prescribed format.

3.1 Machine learning techniques

Several types of machine learning approaches have been used to develop prediction models. Four kinds of ML classifiers are tested, as this study's prediction model is illustrated below.

Decision tree- A decision tree (DT) classifier is a popular classification approach. It is capable of handling enormous volumes of data in both parallel and serial modes. No prior knowledge of factors or domains is required for DT building. Because of its tree-like structure, DT is commonly employed in decision-making processes, with ovals representing leaves and rectangles representing interior nodes. Each internal node has two children and generates child nodes until the subgroup cannot be further divided, producing important information. Internal nodes and leaves represent dataset features and attribute values, respectively, whilst terminal nodes indicate the goal output value.

Random forest Classifier- Random forest is a well-known supervised machine learning technique that may be used to solve both regression and classification issues. It supports both categorical and numerical variables. Random forest builds several decision trees depending on the input data and selects the final output by majority voting of the various decision trees, which is one of its fundamental features. The decision trees are built by randomly selecting a subset of features at each node, which reduces overfitting and improves model accuracy.

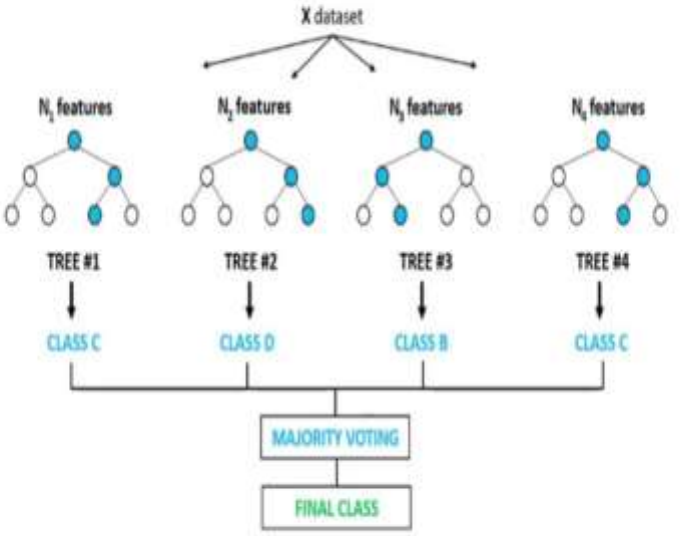

Figure 1: Random Forest classifier.

Extra Trees Classifier-The Extra Trees Classifier is a machine learning algorithm of the ensemble learning family. It is similar to the Random Forest Classifier, however there are some significant distinctions. The Extra Trees Classifier, like the Random Forest, constructs a huge number of decision trees and then combines their predictions to generate a final prediction. Unlike Random Forest, which chooses a subset of features at random to split each node, the Extra Trees Classifier chooses the split point fully at random, with no bias towards any one feature. Furthermore, Extra Trees Classifier frequently uses smaller tree sizes and does not prune the trees, which can result in overfitting if the dataset is too small. When compared to single decision trees, the key advantage of Extra Trees Classifier is that it can reduce variation and overfitting. It is also less susceptible to data noise and can handle a high number of features. However, because the splits are random, the resulting model may be more difficult to comprehend than other models.

AdaBoost- The ensemble learning technique known as AdaBoost, or adaptive boosting, is used in machine learning to solve classification issues. The technique works by combining several weak classifiers to produce a powerful classifier. AdaBoost's core concept is to give each data point in the training dataset a weight, with the misclassified data points receiving a higher weight. On this weighted dataset, it trains a weak classifier and adjusts the weights of the incorrectly categorized data points. Iteratively, this process is repeated, with each weak classifier giving the incorrectly categorized data from the preceding iterations additional weight. In order to create a strong classifier that can precisely categories fresh data points, the weak classifiers are finally integrated. The AdaBoost technique has been extensively utilized in a range of applications, including object detection, face recognition, and speech recognition. It is particularly good at increasing the accuracy of weak classifiers.

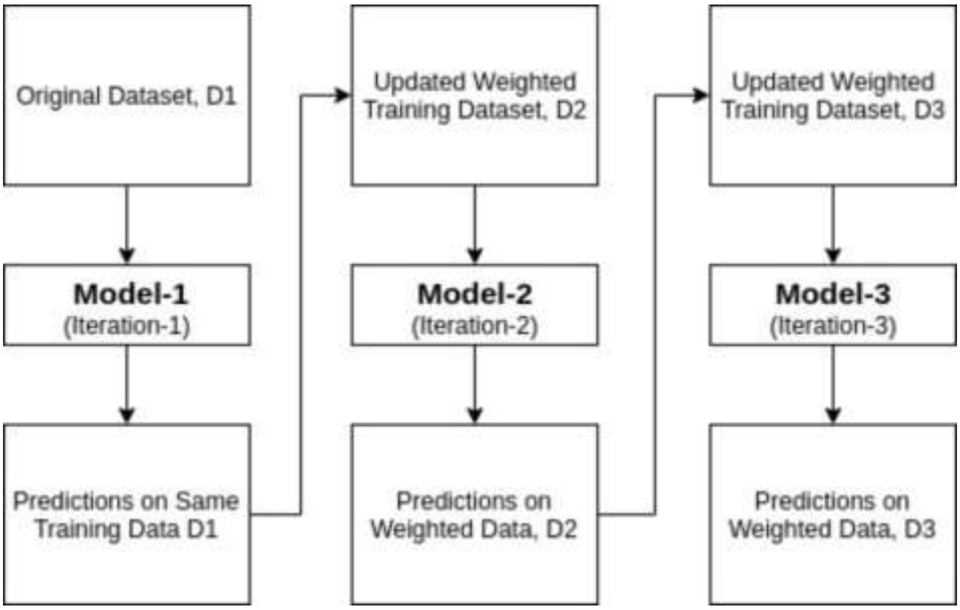
Original Dataset, D1
Updated Weighted Training Dataset, D2
Updated Weighted Training Dataset, D3
Model-1
(Iteration-1)
Model-2
(Iteration-2)
Model-3
(Iteration-3)
Predictions on Same Training Data D1
Predictions on Weighted Data, D2
Predictions on Weighted Data, D3

Figure 2: AdaBoost

K-nearest neighbors (KNN)- An instance-based, non-parametric supervised learning approach is the k-nearest neighbors (KNN) classifier. Regression and classification issues are addressed by it. The fundamental principle of the KNN classifier is to locate the k-nearest neighbors of a new data point using the distance metric and then categorise the data point using the majority class of its k-nearest neighbors. KNN doesn't need any time for training, but it takes a long time to forecast new occurrences, especially if the training data set is big.

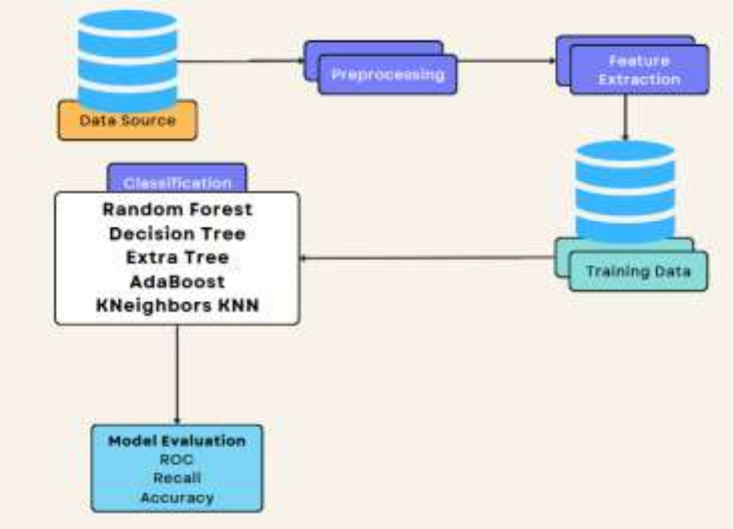

Figure 3: Student Performance Prediction Model

Data Set Collection- The data set used in this study was collected from the Kal Board 360 learning management system and is publicly available on the Kaggle website. The data set includes information on the academic performance of 327 students, of which 181 are female and 146 are male. The data set consists of 17 features, including both academic and non-academic features.

Data Preprocessing- A crucial step in this process is data preparation. Data is gathered from a variety of sources, including databases, Excel files, and log files. Since the extracted data is in a raw format, preparation may not be possible. Inconsistencies, mistakes, and missing values in the data might make it challenging to analyse and create models. Therefore, we must clean and preprocess the data before we analyse it in order to get better findings. This entails eliminating duplicates, adding missing data, addressing outliers, and, if necessary, normalizing or modifying the data. Although the process of preparing data takes time, accurate analysis and modelling depend on it.

Feature Extraction- Once the dataset is cleaned and free of redundancy, we need to preprocess the data. This involves transforming the data into a suitable format for analysis. As mentioned, some classifiers require nominal data, while others require continuous data. Therefore, we may need to use discretization to transform numerical data into the nominal form. Other preprocessing techniques may include scaling the data, dealing with missing values, and handling outliers. The goal is to prepare the data for analysis to ensure accurate and meaningful results.

4. RESULTS AND DISCUSSION

In this section, we aim to predict students' performance in the online learning environment by conducting several experiments on our dataset. We use machine learning classifiers to build a predictive model, and the experiments are conducted using the Jupyter notebook tool. Three different classifiers are used in this study, namely Random Forest, Support Vector Machines (SVM), and Decision Tree. The classifiers are evaluated using the 10-fold cross-validation method, which involves dividing the dataset into 10 subsets of equal size. In each fold, 9 subsets are used for training the model, and the remaining subset is used for testing. This technique helps us to evaluate the performance of our classifiers and to estimate the model's accuracy on unseen data.

Table 2. Result of Implemented Classifier

Classification Technique	Accuracy
Random Forest	0.84
Decision Tree	0.77
Extra tree	0.89
AdaBoost	0.36
KNN	0.70

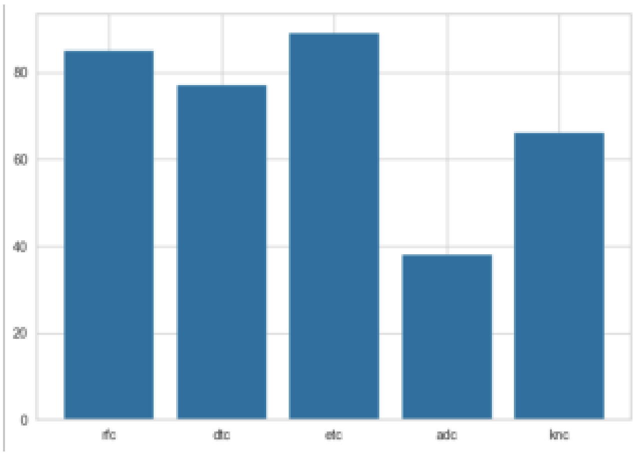

Figure 4: Comparision of Different Classifier

A comparison of the five machine learning classifiers' results is visualized in figure 4 above. Our experiments show promising results with accuracy achieved between 81% and 85%. The implemented classifiers, namely random forest, decision tree, Extra boost, AdaBoost and KNN, have an overall accuracy of 84%, 77%, 89%, 36% and 70%, respectively.

5. CONCLUSION

In conclusion, this project's main goal is to develop a method for assessing student performance. To enable precise prediction of student performance, data mining techniques are used in conjunction with the Support Vector Machine algorithm. The Student Performance Analysis System's (SPAS) main advantage is that it helps lecturers assess students' performance. Lecturers can identify students who are likely to fail a course by using the technique.

6. REFERENCES

[1] J. Xu, K. H. Moon, and M. Van Der Schaar, “A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs,” IEEE J. Sel. Top. Signal Process., vol. 11, no. 5, pp. 742–753, 2017.

[2] K. P. Shaleena and S. Paul, “Data mining techniques for predicting student performance,” in ICETECH 2015 - 2015 IEEE International Conference on Engineering and Technology, 2015, no. March, pp. 0–2.

[3] M. Shahiri, W. Husain, and N. A. Rashid, “A Review on Predicting Student’s Performance Using Data Mining Techniques,” in Procedia Computer Science, 2015.

[4] Y. Meier, J. Xu, O. Atan, and M. Van Der Schaar, “Predicting grades,” IEEE Trans. Signal Process., vol. 64, no. 4, pp. 959–972, 2016.

[5] P. Guleria, N. Thakur, and M. Sood, “Predicting student performance using decision tree classifiers and information gain,” Proc. 2014 3rd Int. Conf. Parallel, Distrib. Grid Comput. PDGC 2014, pp. 126–129, 2015.

[6] J. Shana, and T. Venkatacalam, “A framework for dynamic Faculty Support System to analyse student course data”, International Journal of Emerging Technology and Advanced Engineering, Vol. 2, No. 7, 2012, pp.478-482.