

Understanding Students Performance by Using Machine Learning Techniques

Submitted in partial fulfillment of the requirement for the award of
Bachelor of Engineering degree in Computer Science and Engineering

by

ABHINAY SARVESWARA RAO GRANDHI (Reg.No - 40110013)

AKULA DATTA VEERA SASANK MITRA (Reg.No - 40110050)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

CATEGORY – 1 UNIVERSITY BY UGC

Accredited with Grade “A++” by NAAC | 12B Status by UGC | Approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHISALAI, CHENNAI – 600119

APRIL – 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Category - I University by UGC

Accredited "A++" by NAAC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Abhinay Sarveswara Rao Grandhi (40110013)** and **Akula Datta Veera Sasank Mitra (40110050)** who carried out the project work entitled **“UNDERSTANDING STUDENTS PERFORMANCE BY USING MACHINE LEARNING TECHNIQUES ”** under my supervision from November 2023 to April 2024.

Internal Guide

Dr. R. SETHURAMAN, B.E., M.E., Ph.D.

Head of the Department

Dr. L. LAKSHMANAN, B.E., M.E., Ph.D.

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I, **Akula Datta Veera Sasank Mitra (40110050)** , hereby declare that the Project Work entitled “**Understanding Students Performance by using Machine Learning Techniques**” done by me under the guidance of **Dr. R. Sethuraman, B.E., M.E., Ph.D.**, is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

DATE:

PLACE: Chennai

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to The **Board of Management of Sathyabama Institute of Science and Technology** for their encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala, M.E., Ph.D., Dean**, School of Computing and **Dr. L. Lakshmanan, M.E., Ph.D.**, Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. R. Sethuraman, B.E., M.E., Ph.D.**, for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

In many organizations, data mining techniques are used for analyzing large amount of available data's, information's for decision making process. In educational sector, Data mining is used for wide variety of application's such as performance of the students like mark, attendance, staff opinion, extracurricular activities, Ragging and stress. The data mining techniques used for identifying the performance of the student using K-means and KNN algorithms. These two algorithm's identifies and analyses the performance of the student. Subsequently, data pre-processing takes center stage, entailing data cleansing and feature engineering. Missing data is systematically addressed, outliers are carefully managed, and the dataset is meticulously prepared for in-depth analysis. Upon data preparation, the dataset is partitioned into training and testing sets, ensuring a robust evaluation of the machine learning algorithms' performance on unseen data, a vital step in establishing their generalization capabilities. The results of this study promise to provide actionable insights to enhance student performance, influencing strategies for personalized support, targeted resources, and pedagogical adjustments.

Ethical considerations, including data privacy and fairness, are pivotal aspects of this research, reaffirming the commitment to responsible and equitable educational advancement. This project delves into the realm of machine learning to gain deeper insights into students' academic performance. By employing a spectrum of machine learning algorithms, including regression, classification, and clustering techniques, the research endeavors to unearth pivotal predictors of student achievement, empowering educators with actionable insights to tailor interventions and support mechanisms. Furthermore, this investigation seeks to probe beyond traditional academic metrics, delving into the potential influence of non-academic variables such as extracurricular engagements, familial dynamics, and individual interests on students' scholastic outcomes. Ultimately, the findings derived from this endeavor hold the promise of informing educational policies and practices, paving the way for a more personalized and efficacious approach to student support and intervention strategies that cater to the holistic needs of learners.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	v
	LIST OF FIGURES	viii
	LIST OF ABBREVIATIONS	ix
1	INTRODUCTION	1
2	LITERATURE SURVEY	2
	2.1 Inferences From Literature Survey	2
	2.2 Existing System	4
3	AIM AND SCOPE OF THE PRESENT INVESTIGATION	5
	3.1 AIM	5
	3.2 SCOPE OF THE PRESENT INVESTIGATION	5
	3.3 FEASIBILITY STUDIES AND RISK ANALYSIS	5
	3.4 SYSTEM REQUIREMENTS SPECIFICATION	6
4	EXPERIMENTAL METHODS, ALGORITHMS USED	7
	4.1 PROPOSED SYSTEM	7
	4.2 ALGORITHMS USED	7
	4.3 SYSTEM ARCHITECTURE	9
	4.4 UML DIAGRAM	10
5	IMPLEMENTATION DETAILS	12
	5.1 IMPLEMENTATION	12
	5.2 MODULES	13
	5.3 SOFTWARE IMPLEMENTATION PLAN	13
	5.4 TESTING PLAN	14
6	RESULTS AND DISCUSSIONS, PERFORMANCE ANALYSIS	15
	6.1 RESULTS AND DISCUSSIONS	15
	6.2 SAMPLE SCREENSHOTS	16

7	SUMMARY AND CONCLUSION	17
	7.1 SUMMARY	17
	7.2 CONCLUSION	17
	REFERENCES	18
	APPENDIX	19
	A. SOURCE CODE	19
	B. SAMPLE SCREENSHOTS	47
	C. RESEARCH PAPER	48
	D. CERTIFICATE	54

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
4.1	K-MEANS CLUSTERING ALGORITHM	7
4.2	SYSTEM ARCHITECTURE	9
4.3	SEQUENCE DIAGRAM	11
4.4	COLLABORATION DIAGRAM	12
B.1	DATA ENTRY PAGE	47
B.2	ACCURACY DIAGRAM	47

LIST OF ABBREVIATIONS

ABBREVIATION	EXPANSION
IDE	INTEGRATED DEVELOPMENT ENVIRONMENT
KNN	K-NEAREST NEIGHBOR
UML	UNIFIED MODELING LANGUAGE

CHAPTER 1

INTRODUCTION

There are many tremendous improvement research interests in using machine learning in educational sector. This modern emerging sector, called educational data mining, concerned with improved methods that extract knowledge from data come from the educational sector. machine learning is a technique of sorting which is actually used to extract hidden patterns from huge databases. This concepts and methods can be applied in various fields like marketing, medicine, real estate, customer relationship management, engineering, web mining, etc. Educational machine learning is a new emerging or advanced technique of machine learning that can be applied on the data related to the field of education. The data can be collected from past used data and operational data reside in the databases of educational institutes. The data of students can be personal information or academic performance. Furthermore it can be achieve from e learning database systems which have a huge amount of data and information used by most institutes .

It uses many techniques for proper implementation of machine learning concepts such as K-means Clustering and K-Nearest neighbour. Using these techniques different kinds of knowledge can be discovered using association rules, classification and clustering. By using this we extract knowledge that describes students performance in examination and all their detail information. From This huge amounts of data, the first task is to sort them out, cluster analysis is to classify the raw data in a reasonable way. Clustering is a bunch of physical or abstract objects, as per the degree of similarity between them, divided into several groups, and makes the same data objects within a groups of high similarity and different groups of data objects which are not similar.

Aims to utilize various machine learning algorithms to analyze and predict students academic performance based on relevant features such as socio-economic background, study habits, and demographic information. By applying machine learning models to a dataset containing information about students' performance and characteristics, the project seeks to uncover patterns and insights that can help educators and policymakers make informed decisions to enhance educational outcomes.

CHAPTER 2

LITERATURE SURVEY

2.1 INFERENCES FROM LITERATURE SURVEY

2.1.1 Prediction of Students Academic Performance using Data Mining: Analysis

Author/Year: Sayana T S/ 2018

Techniques: Association rule algorithm, K-means clustering algorithm and Decision tree

Results: In this paper, there is a evaluation between students association rule algorithm, K-means clustering algorithm and Decision tree. This survey for assessing students performance is based on different attributes. Attributes include class quizzes, mid and final exam marks, assignment, lab works etc. We discuss the procedure based on Decision Tree of data mining method, Data Clustering that enable academicians to predict students performance and instructor can take essential step to develop student academic performance.

Advantages: The efficiency of various decision tree algorithms can be analyzed based on their accuracy and time taken to derive the tree. The predictions obtained from the system have helped the tutor to identify the weak students and improve their Performance. The analysis of the result declared from the university is a proof for the same. Since the application of data mining brings a lot of advantages in higher learning institution.

Disadvantages: That is the number of cluster, must be determined earlier. It is difficult to determine the number of clusters when there is slight change in the data.

In this Prediction of Students Academic Performance using Data Mining: Analysis project we have considered both advantages and disadvantages. So to make sure that understanding the pros and cons of a project.

K-means clustering algorithm and Decision tree there is a evaluation between students association rule algorithm. So we can improve the performance of our project by doing this.

2.1.2 Predicting Student Performance Using Data Mining and Learning Analytics

Techniques: A Systematic Literature Review Author/Year:

Abdallah Namoun and Abdullah Alshantiti/ 2020 .

TECHNIQUES: Artificial Neural Networks, Support Vector Machine, KNearest Neighbor, and Random Forests.

Results: The best practices for conducting systematic literature reviews, e.g., PICO and PRISMA, were applied to synthesize and report the main results. The attainment of learning outcomes was measured mainly as performance class standings (i.e., ranks) and achievement scores (i.e., grades). Regression and supervised machine learning models were frequently employed to classify student performance. Finally, student online learning activities, term assessment grades, and student academic emotions were the most evident predictors of learning outcomes. We conclude the survey by highlighting some major research challenges and suggesting a summary of significant recommendations to motivate future works in this field.let us see advantages and disadvantages.

Advantages: Identified that attributes and methods that predict student performance
Applied the PICO methodology.

Disadvantages: These reviews suffered from several limitations, for they were generally broad, did not focus on using student outcomes as an indicator of student performance, suffered from quality issues (e.g., methodologies not thoroughly defined, and were not published in highly indexed venues.

2.1.3 Student Performance Prediction Using Data Mining Techniques

Author/Year: Urgesh Ugale, Jeet Pawar, Sachin Yadav, Dr. Chandrashekhar Raut/ 2020.

TECHNIQUES: Decision Tree, Naive Bayes, Support Vector Machine, K-Nearest Neighbors .

Results: In this approach, preprocessing step will be applied to raw dataset so that the mining algorithm will be applied properly.The prediction about students" performance can help him/her to enhance the performance.

Advantages: Did not address student outcome from different perspectives – Did not discuss the limitations – Did not compare the predictive models.

2.1.4 Student Performance Prediction Model based on Supervised Machine Learning Algorithms

Author/Year: Ali Salah Hashim, Wid Akeel Awadh, Alaa Khalaf Hamoud/ 2020.

TECHNIQUES: Decision Tree, Naïve Bayes, Logistic Regression, Support Vector Machine, K-Nearest Neighbour .

Results: In this study, we compared the performances of several supervised machine learning algorithms. Results indicated that logistic regression classifier is the most accurate in predicting the exact final grades of students (68.7% for passed and 88.8% for failed).

Advantages: Listed the techniques used in the learning analytics for predicting student performance .

Disadvantages: Did not assess the quality of the studies – Did not compare the model.

2.2 EXISTING SYSTEM

- As of now, Existing system take only performance into consideration which is not sufficient for having system, which can help us to evaluate performance of a student.
- We are not having a system which would help us to integrate the performance and undesirable into consideration.

Problems

- Existing system miss the undesirable data for the students.
- Data Analysis: If applicable, analyze data related to the system's performance, user behavior, and outcomes. Data can provide valuable insights into problems and It may not check the social data for the student.
- User Feedback: Collect user feedback through surveys, feedback forms, or interviews. Understand what users like and dislike about the system and any issues they encounter.
- Models trained on one educational setting may not generalize well to others.
- Lack of adaptability limits practical use, Irrelevant or redundant features affect model performance.
- In educational datasets, the number of students who drop out is often smaller than those who stay.

CHAPTER 3

AIM AND SCOPE OF THE PRESENT INVESTIGATION

3.1 AIM

The main is to improve performance of a student by studying the factors that effect student performance. Using machine learning techniques is typically to develop predictive models that can forecast academic outcomes based on various factors. These factors might include demographic data, academic history, engagement levels, and more

3.2 SCOPE OF THE PRESENT INVESTIGATION

The scope of the present investigation on understanding students' performance using machine learning techniques encompasses a multifaceted approach to unraveling the complex dynamics that influence academic outcomes. It involves analyzing a diverse range of factors such as socio-economic background, demographic characteristics, study habits, and classroom environment to gain deeper insights into the predictors of student success. The investigation aims to not only develop accurate predictive models but also to interpret the underlying factors driving performance predictions. Additionally, the scope extends to addressing ethical considerations, ensuring fairness, transparency, and accountability in the deployment of machine learning algorithms within educational settings.

3.3 FEASIBILITY STUDIES AND RISK ANALYSIS

Feasibility studies are conducted to assess the practicality and viability of a project, business idea, or initiative before committing significant resources to it. They typically involve a comprehensive analysis of various aspects to determine whether the project is worth pursuing. Here are some key elements of a feasibility study:

- **Market Feasibility:** This assesses the potential demand for the product or service in the target market. It involves market research, competitor analysis, and understanding customer needs and preferences.
- **Technical Feasibility:** Evaluates the technology, infrastructure, and resources required to implement the project. It determines if the necessary technology is available or if it can be developed or acquired.

- **Financial Feasibility:** Examines the financial aspects of the project, including cost estimation, revenue projections, and profitability. It also considers potential sources of funding and investment returns.
- **Operational Feasibility:** Assesses the practicality of implementing the project from an operational perspective. This includes evaluating processes, logistics, and resources required for day-to-day operations.
- **Legal and Regulatory Feasibility:** Investigates the legal and regulatory requirements and constraints that might affect the project.
- **Environmental Feasibility:** Considers the environmental impact of the project and assesses its sustainability. This is particularly important for projects that could have significant environmental implications.
- **Schedule Feasibility:** Examines whether the project can be completed within the desired time frame. It involves creating a project timeline and identifying potential bottlenecks.
- **Resource Feasibility:** Evaluates the availability and allocation of resources, such as human resources, materials, and equipment, to complete the project.
- **Risk Analysis:** Identifies potential risks and uncertainties associated with the project and develops mitigation strategies to address them.

3.4 SYSTEM REQUIREMENTS SPECIFICATION DOCUMENT HARDWARE REQUIREMENTS

- System - Pentium-IV
- Speed - 2.4GHZ
- Harddisk - 40GB
- Monitor - 15VGA color
- RAM - 512M

Software Requirements

- Operating System - Windows XP
- Coding language - Java
- IDE - Net beans

CHAPTER 4

EXPERIMENTAL AND METHODS: ALGORITHMS USED

4.1 PROPOSED SYSTEM

- The work aims to develop a trust model using data mining techniques, which mines required information, so that the present education system may adopt this as a strategic management tool.
- The proposed system use educational data mining techniques to evaluate performance and identify undesirable behavior.
- In educational sector, Data mining is used for wide variety of application's such as performance of the students like mark, attendance, staff opinion, extracurricular activities, Ragging and stress. The data mining techniques used for identifying the performance of the student using K-means and KNN algorithms.

4.2 ALGORITHMS USED

4.2.1 K-means clustering algorithm

What is K-means?

1. Partitional clustering approach
2. Each cluster is associated with a centroid (center point)
3. Each point is assigned to the cluster with the closest centroid
4. Number of clusters K must be specified

Input: k (the number of clusters),
 D (a set of lift ratios)
Output: a set of k clusters
Method:
Arbitrarily choose k objects from D as the initial cluster centers;
Repeat:
 1. (re)assign each object to the cluster to which the object is
 the most similar, based on the mean value of the objects in the
 cluster;
 2. Update the cluster means, i.e., calculate the mean value of
 the objects for each cluster
Until no change;

FIG 4.1: K-Means clustering algorithm

Details of K-means

1. Initial centroids are often chosen randomly. - Clusters produced vary from one run to another.
2. The centroid is (typically) the mean of the points in the cluster.
3. 'Closeness' is measured by Euclidean distance, cosine similarity, correlation, etc.
4. K-means will converge for common similarity measures mentioned above.
5. Most of the convergence happens in the first few iterations. - Often the stopping condition is changed to 'Until relatively few points change clusters.'

4.2.2 K-Nearest Neighbour(KNN) Classification Method

K-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The k-NN algorithm is among the simplest of all machine learning algorithms. The neighbors are taken from a set of objects for which the class (for K-NN classification) or the object property value (for K-NN regression) is known.

STEP 1: BEGIN

STEP 2: Input: $D = \{(x_1, c_1), \dots, (x_N, c_N)\}$.

STEP 3: $x = (x_1, \dots, x_n)$ new instance to be classified

STEP 4: FOR each labelled instance (x_i, c_i) calculate $d(x_i, x)$

STEP 5: Order $d(x_i, x)$ from lowest to highest, $(i = 1 \dots N)$

STEP 6: Select the K nearest instances to x: D_x^K

STEP 7: Assign to x the most frequent class in D_x^K

STEP 8: END.

K-Means clustering is an unsupervised learning algorithm. There is no labeled data for this clustering, unlike in supervised learning. K-Means performs the division of objects into clusters that share similarities and are dissimilar to the objects belonging to another cluster. The term 'K' is a number K-means clustering is frequently used in data analysis, and a simple example with five x and y value pairs to be placed into two clusters using the Euclidean distance.

KMeans is used across many fields in a wide variety of use cases; some examples of clustering use cases include customer segmentation, fraud detection, predicting account attrition, targeting client incentives, cybercrime identification, and delivery route optimization.

4.3 SYSTEM ARCHITECTURE

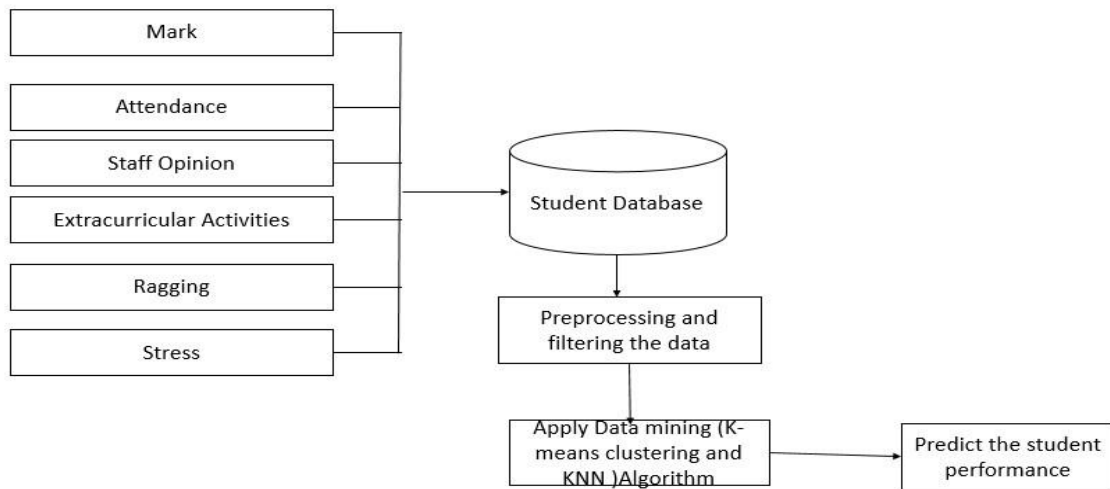


FIG 4.2: Architecture Diagram

The system architecture for understanding students' performance using machine learning techniques encompasses several interconnected components designed to facilitate data processing, analysis, and interpretation. At its core, the architecture includes data collection modules that gather student records, assessments, and demographic information from various sources. Preprocessing modules clean, normalize, and transform the data to prepare it for analysis. Machine learning models, including algorithms like K-Nearest Neighbors (KNN) and K-Means clustering, are applied to the preprocessed data to uncover patterns and insights into student performance. Interpretation modules visualize and interpret the results, providing actionable insights for educators and policymakers. Ethical considerations are integrated into the architecture to ensure data privacy, fairness, and transparency throughout the process. The system architecture facilitates seamless collaboration between stakeholders and enables the effective utilization of machine learning techniques to enhance educational outcomes and student success. The system architecture for understanding students' performance using machine learning techniques involves collecting and preprocessing data on various academic and demographic factors. Feature engineering is then applied to extract relevant features, followed by the selection and training of appropriate machine learning models. Continuous monitoring and maintenance ensure the system's effectiveness over time, while prioritizing privacy and security safeguards to protect sensitive student information.

4.4 UML DIAGRAMS

4.4.1 Sequence Diagram

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

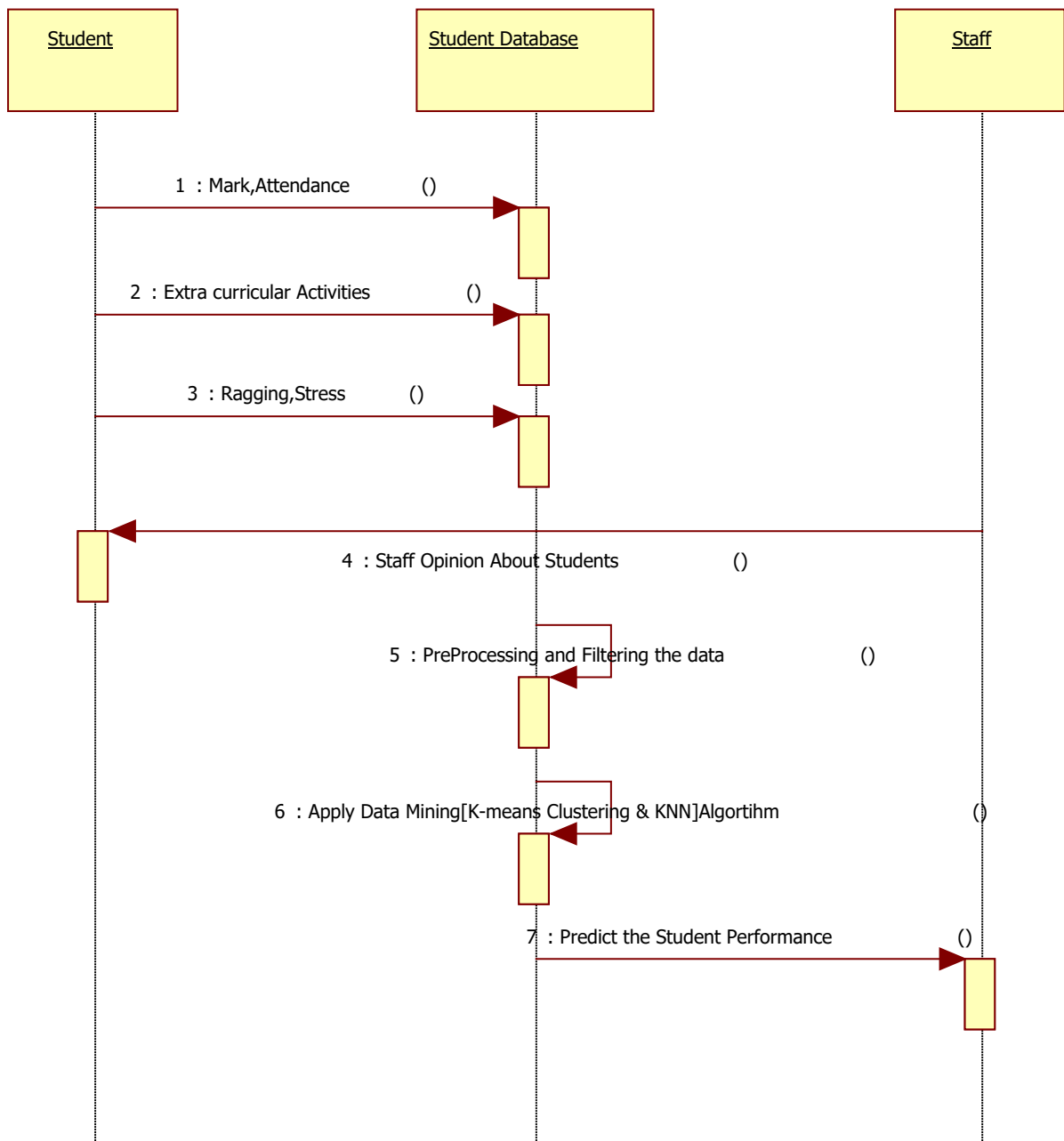


FIG 4.3: Sequence Diagram

4.4.2 Collaboration Diagram

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). Developers can use these diagrams to portray the dynamic behavior of a particular use case and define the role of each object.

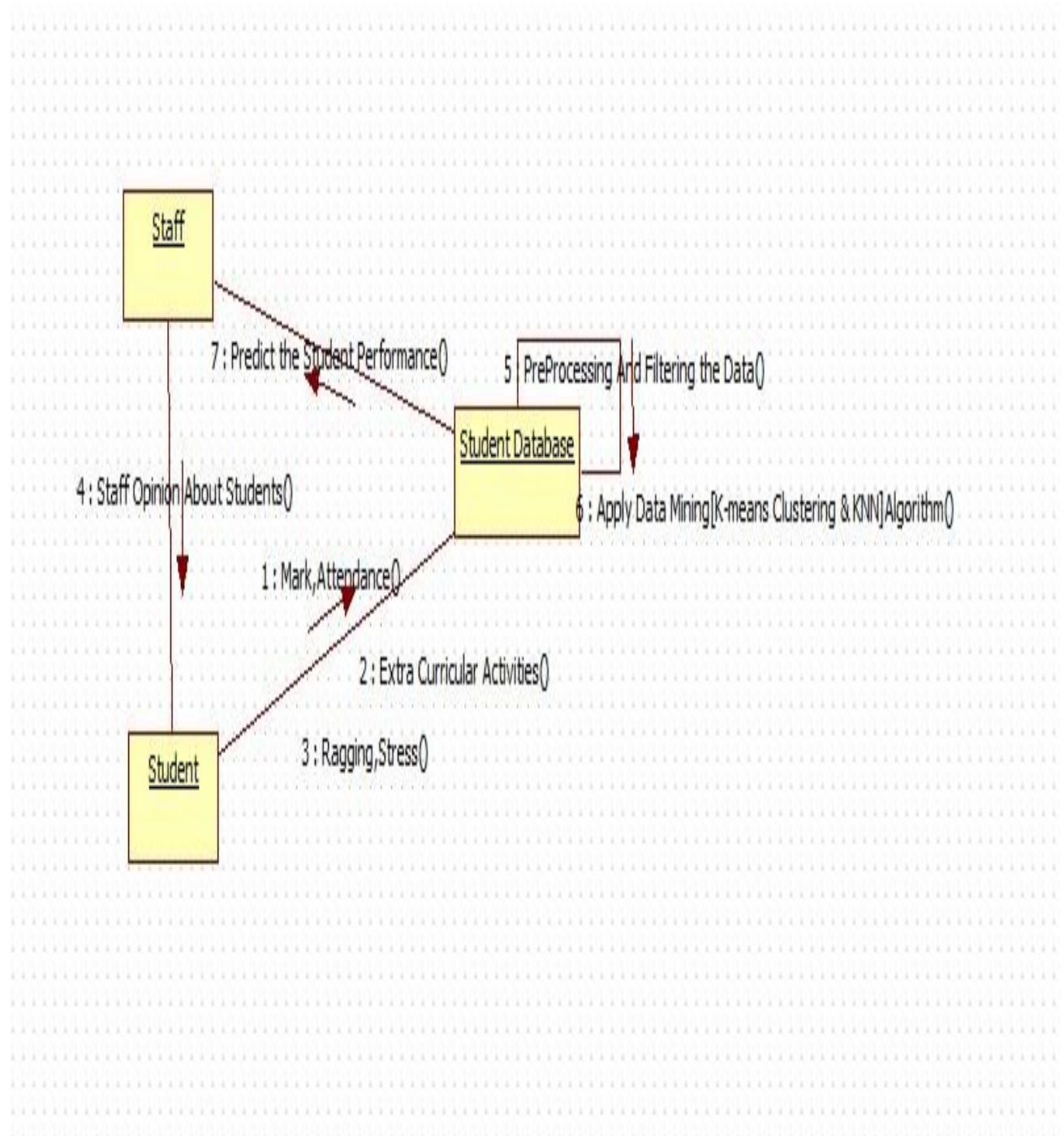


FIG 4.4: Collaboration Diagram

CHAPTER 5

IMPLEMENTATION DETAILS

5.1 IMPLEMENTATION

Implementing a project to understand students' performance using machine learning techniques involves several key steps and considerations:

- **Data Collection:** Gather data from various sources such as student records, surveys, and academic assessments. Ensure that the data collected is comprehensive and relevant to the research question.
- **Data Preprocessing:** Clean the data by handling missing values, outliers, and inconsistencies. Perform data transformations such as normalization or standardization to prepare the data for analysis.
- **Exploratory Data Analysis (EDA):** Conduct exploratory data analysis to understand the distribution of variables, identify patterns, correlations, and potential relationships between features and the target variable (student performance).
- **Feature Engineering:** Engineer new features or transform existing ones to better represent underlying patterns in the data. This step can involve techniques such as one-hot encoding, feature scaling, and dimensionality reduction.
- **Model Selection:** Experiment with different machine learning algorithms suitable for the task, such as linear regression, decision trees, random forests, support vector machines, or neural networks. Choose the most appropriate models based on their performance and interpretability.
- **Model Training:** Split the data into training and testing sets, and train the selected models on the training data. Utilize techniques like cross-validation and hyperparameter tuning to optimize model performance.
- **Model Evaluation:** Evaluate the trained models using appropriate evaluation metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. Compare the performance of different models to select the best-performing one..
- **Ethical Considerations:** Address ethical considerations such as data privacy, fairness, and bias.
- These all were consider while implementation of our model .

5.2 MODULES

- Data Collection
- Preprocessing
- Classification module
- Prediction

Data Collection

In this module, Student data's will be collected from the college. Student's data like mark, attendance, staff opinion, Social media, extracurricular activities, Ragging and stress.

Preprocessing

Data pre-processing is done to remove the incomplete noisy and inconsistent data. Data must be pre-processed before using in feature selection task.

Classification Module

The machine learning techniques used for identifying the performance of the student using Naïve Bayes and KNN algorithms. These two algorithm's identifies and analyses the performance of the student.

Prediction

In this module, to predict the student performance based upon student mark, attendance, staff opinion, Social Media, extracurricular activities, Ragging and stress.

5.3 SOFTWARE IMPLEMENTATION PLAN

Overview: Begin with an overview of the software implementation plan, outlining the purpose and goals of the plan.

Development Environment: Describe the software development environment, including programming languages, frameworks, and tools to be used.

System Architecture: Provide an overview of the proposed system's architecture, including components, modules, and their interactions.

Database Design: Explain the structure of the database if applicable, including data models, tables, and relationships.

Algorithms and Models: Describe the machine learning algorithms, models, or methodologies that will be implemented to understand student performance.

Data Sources: Specify the data sources that will be used for training and testing the model, including data collection methods.

Implementation Timeline: Create a timeline with milestones and deadlines for the software development process.

5.4 TESTING PLAN

Testing Objectives: Clearly state the objectives of testing, such as verifying the correctness of the system, identifying and fixing defects, and ensuring that it meets user requirements.

Test Scenarios: Define test scenarios, which are high-level descriptions of what will be tested. For example, "Test the accuracy of student performance predictions."

Test Cases: Break down test scenarios into detailed test cases with steps to follow, expected outcomes, and criteria for success.

Types of Testing: Describe various types of testing to be performed, including unit testing, integration testing, system testing, and user acceptance testing (UAT).

Testing Data: Specify the data that will be used for testing, including sample datasets and expected results.

Testing Tools: List any testing tools or frameworks that will be used for automated testing, if applicable.

Testing Schedule: Create a testing schedule that aligns with the implementation timeline, including specific dates for each type of testing.

Bug Tracking and Reporting: Define procedures for reporting and tracking issues or bugs identified during testing.

User Acceptance Testing: Outline the process for involving end-users or stakeholders in UAT to ensure that the system meets their requirements and expectations.

Regression Testing: Describe how regression testing will be conducted to ensure that new changes do not introduce new defects.

Performance Testing: If relevant, detail how performance testing will be carried out to assess the system's response times and scalability.

Security Testing: If applicable, discuss security testing methodologies to identify and address potential vulnerabilities.

The testing plan for understanding students' performance using machine learning techniques plays a crucial role in ensuring the reliability, accuracy, and effectiveness of predictive models deployed in educational settings

Moreover, interpretability techniques facilitate the interpretation of model predictions, providing actionable insights for educators and policymakers.

CHAPTER 6

RESULTS AND DISCUSSIONS

6.1 RESULTS AND DISCUSSIONS

Several studies, such as those conducted by Pandey and Pal, Hijazi and Naqvi, and Khan, have examined the correlation between various factors and student performance. Pandey and Pal utilized Bayes Classification to predict the performance of new students based on categories, language, and background qualifications. Similarly, Hijazi and Naqvi identified factors such as students' attitudes towards attendance, study hours, family income, and maternal education as significantly related to student performance, employing simple linear regression analysis. Khan's study delved into the prognostic value of cognition, personality, and demographic variables on academic success, highlighting gender and socioeconomic status as influential factors.

The efficacy of data mining techniques in educational contexts is underscored by their ability to classify databases and evaluate student performance. K-means clustering, as a partitional clustering approach, partitions data into distinct clusters associated with centroids, facilitating the identification of patterns and trends. These techniques not only aid in performance evaluation but also serve as invaluable tools in identifying undesirable behaviors, contributing to a holistic understanding of student dynamics within educational ecosystems.

Data preprocessing plays a critical role in the data analysis pipeline, guaranteeing well-structured and consistent data devoid of anomalies. Techniques such as imputation and outlier management are employed to address missing values and prevent skewed analysis.

Imagine you have data about students: their grades, attendance, study habits, and more.

Apply K-means to group students based on similarities.

For example:

Cluster 1: High achievers who attend classes regularly.

Cluster 2: Struggling students with irregular attendance.

Cluster 3: Average performers who study consistently.

6.2 PERFORMANCE ANALYSIS

- Exploration: We will embark on an exploratory journey through the landscape of data-driven education. We will examine the sources of data, delve into the methodologies used for data analysis, and investigate the machine learning algorithms leveraged for understanding student performance.
- Insight: With each step of this exploration, we aim to unearth actionable insights that empower educators, institutions, and policymakers to make data-informed decisions. These insights can aid in the identification of at risk students, the personalization of learning experiences, and the continuous improvement of education.
- Transformation: Ultimately, this presentation seeks to illuminate how the fusion of data-driven approaches, machine learning, and education has the potential to transform the educational landscape.

Performance analysis using machine learning techniques for understanding students' performance involves a multifaceted approach to extract meaningful insights from student data. Beginning with comprehensive data collection from various sources such as student records, assessments, and demographic information, the analysis proceeds to meticulous preprocessing steps to ensure data quality and consistency. Subsequent feature engineering techniques enhance the dataset's predictive power, capturing essential aspects of student performance. Machine learning models, ranging from regression to classification algorithms, are then applied to the prepared dataset to uncover patterns and relationships between student attributes and academic outcomes.

Model evaluation metrics such as accuracy, precision, recall, and F1-score are employed to assess the models' performance and generalization capabilities. Additionally, interpretability techniques like feature importance analysis and model visualization aid in understanding the factors influencing student performance. The insights gained from performance analysis inform evidence-based interventions and strategies tailored to meet the diverse needs of student. Simultaneously, the K-Means clustering algorithm is utilized to group students into clusters based on similarities in their academic attributes and behavior patterns. The optimal number of clusters is determined using techniques like the elbow method or silhouette score.

CHAPTER 7

SUMMARY AND CONCLUSION

7.1 SUMMARY

The study encompasses a comprehensive exploration of utilizing machine learning and data mining techniques in the realm of higher education, with a focus on understanding student performance and behaviors. Through a synthesis of various research efforts, the efficacy and potential of these methodologies are elucidated.

The construction of a trust model utilizing data mining techniques emerges as a focal point in the present study. Educational databases serve as repositories of vital information, encompassing various metrics such as grades, attendance, staff feedback, and extracurricular activities. Through the utilization of data mining algorithms like K-means clustering and K-nearest neighbor (KNN), educators can extract valuable insights into student behaviors and academic outcomes.

The proposed system, with its emphasis on integrating data mining techniques into the educational framework, offers several advantages. Educational databases contain a wealth of information crucial for evaluating student performance, and data mining techniques serve as invaluable tools in classifying this information and extracting meaningful insights.

7.2 CONCLUSION

In conclusion, the development of a trust model utilizing data mining techniques represents a significant step towards enhancing the efficacy and efficiency of the education system. By leveraging methodologies such as K-means clustering and KNN classification, educators can gain deeper insights into student performance and behaviors, fostering a conducive learning environment. Moreover, meticulous data collection and preprocessing ensure the integrity and reliability of the data, laying the groundwork for robust analysis and model development. Ultimately, the integration of data mining into the educational framework holds immense potential for driving positive outcomes and fostering student success. In future endeavors focused on understanding students' performance using machine learning techniques, several promising avenues beckon for exploration and innovation. styles and extracurricular activities could provide deeper insights into academic outcomes.

REFERENCES

- [1] A. C. Graesser, X. Hu, and R. Sottolare, "Intelligent tutoring systems," in *Proc. Int. Handbook Learn. Sci.*, 2018, pp. 246–255.
- [2] A. Banumathi, A. Pethalakshmi. A novel approach for upgrading Indian education by using data mining techniques.
- [3] A. Dutt, M. A. Ismail, and T. Herawan, "A systematic review on educational data mining," *IEEE Access*, vol. 5, pp. 15991–16005, 2017.
- [4] Anoopkumar M ; A. M. J. Md. Zubair Rahman, A Review on Data Mining techniques and factors used in Educational Data Mining to predict student amelioration.
- [5] B. Xu, S. Yan, X. Jiang, and S. Feng, "SCFH: A student analysis model to identify students' programming levels in online judge systems," *Symmetry*, vol. 12, p. 601, Feb. 2020, doi: 10.3390/sym12040601.
- [6] E. Osmanbegovic, Mirza Suljic. Data Mining Approach For Predicting Student Performance, Using data mining the aim was to develop a model which can derive the conclusion on student's academic success.
- [7] Kin Fun Li, David Rusk and Fred Song. Predicting Student Academic Performance, The performance predictors, if identified, can then be used effectively to formulate corrective action plans to improve the attrition rate.
- [8] M. A. Revilla, S. Manzoor, and R. Liu, "Competitive learning in informatics: The UVA online judge experience," *Olympiads Inf.*, vol. 2, pp. 131–148, Feb. 2008.
- [9] M. Durairaj, C. Vijitha. Educational Data mining for Prediction of Student performance Using Clustering Algorithms. The data mining techniques are more helpful in classifying educational database Which contain the useful information for predicting a student's performance.
- [10] T. Saito and Y. Watanobe, "Learning path recommendation system for programming education based on neural networks," *Int. J. Distance Educ. Technol.*, vol. 18, no. 1, pp. 36–64, 2020.
- [11] Y. Watanobe, M. M. Rahman, U. K. Rage, and R. Penugonda, "Online automatic assessment system for program code: Architecture and experiences," in *Advances and Trends in Artificial Intelligence*, vol. 12799, H. Fujita, A. Selamat, J. Lin, M. Ali, Eds. Cham, Switzerland: Springer, Jul. 2021, pp. 272–283.

APPENDIX

A. SOURCE CODE

Admin.java

```
import javax.swing.JDialog;
import javax.swing.JOptionPane;

public class Admin extends javax.swing.JFrame {
    public Admin() {
        initComponents();
        setLocationRelativeTo(null);
    }
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jLabel2 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton4 = new javax.swing.JButton();
        jButton5 = new javax.swing.JButton();
        jButton6 = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jPanel1.setLayout(null);

        jLabel2.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/adminlogin.png"))); // NOI18N
        jPanel1.add(jLabel2);
        jLabel2.setBounds(0, 0, 1010, 290);
```

```

jButton1.setBackground(new java.awt.Color(153, 204, 255));
jButton1.setFont(new java.awt.Font("Times New Roman", 1, 30)); // NOI18N
jButton1.setForeground(new java.awt.Color(255, 255, 255));
jButton1.setText("View Student Data");
jButton1.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 51, 204)));
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});
jPanel1.add(jButton1);
jButton1.setBounds(500, 440, 340, 80);

```

```

jButton2.setBackground(new java.awt.Color(153, 204, 255));
jButton2.setFont(new java.awt.Font("Times New Roman", 1, 30)); // NOI18N
jButton2.setForeground(new java.awt.Color(255, 255, 255));
jButton2.setText("Remove Student Data");
jButton2.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 51, 204)));
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});
jPanel1.add(jButton2);
jButton2.setBounds(90, 440, 350, 80);

```

```

jButton4.setBackground(new java.awt.Color(153, 204, 255));
jButton4.setFont(new java.awt.Font("Times New Roman", 1, 30)); // NOI18N
jButton4.setForeground(new java.awt.Color(255, 255, 255));
jButton4.setText("Create Mark Sheet");

```

```

        jButton4.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 51, 204)));
        jButton4.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton4ActionPerformed(evt);
            }
        });
        jPanel1.add(jButton4);
        jButton4.setBounds(500, 320, 340, 80);
        jButton5.setBackground(new java.awt.Color(153, 204, 255));
        jButton5.setFont(new java.awt.Font("Times New Roman", 1, 30)); // NOI18N
        jButton5.setForeground(new java.awt.Color(255, 255, 255));
        jButton5.setText("Add Student Data");
        jButton5.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 51, 204)));
        jButton5.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton5ActionPerformed(evt);
            }
        });
        jPanel1.add(jButton5);
        jButton5.setBounds(90, 320, 350, 80);

        jButton6.setBackground(new java.awt.Color(153, 204, 255));
        jButton6.setFont(new java.awt.Font("Times New Roman", 1, 30)); // NOI18N
        jButton6.setForeground(new java.awt.Color(255, 255, 255));
        jButton6.setText("Prediction");
        jButton6.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 51, 204)));
        jButton6.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton6ActionPerformed(evt);
            }
        });

```

```

});
jPanel1.add(jButton6);
jButton6.setBounds(310, 560, 340, 80);

jLabel1.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/secure.jpg"))); // NOI18N
jLabel1.setText("jLabel1");
jPanel1.add(jLabel1);
jLabel1.setBounds(0, 0, 1500, 710);

javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap()
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, 1502,
Short.MAX_VALUE)
            .addGap())
        );
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap()
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, 709,
Short.MAX_VALUE)
            .addGap())
        );

pack();
} // </editor-fold>

```

```

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {
    {
        setVisible(false);

        HomePage al = new HomePage();
        al.setVisible(true);
    }
}

```

```

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
    setVisible(false);
    StudentForm sf = new StudentForm ();
    sf.setVisible(true);
}

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    setVisible(false);
    ViewStudentData vs = new ViewStudentData ();
    vs.setVisible(true);
}

```

```

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
    setVisible(false);
    StudentMarkSheet sms =new StudentMarkSheet();
    sms.setVisible(true);
}

```

```

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    setVisible(false);
    RemoveStudent rs =new RemoveStudent();
    rs.setVisible(true);
}

```

```

    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
            javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Admin.class.getName()).log(java.util.logging.Level
    .SEVERE, null, ex);

        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Admin.class.getName()).log(java.util.logging.Level
    .SEVERE, null, ex);

        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Admin.class.getName()).log(java.util.logging.Level
    .SEVERE, null, ex);

        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Admin.class.getName()).log(java.util.logging.Level
    .SEVERE, null, ex);

        }
    }
}

//</editor-fold>

    java.awt.EventQueue.invokeLater(() -> {
        new Admin().setVisible(true);
    });
}

private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton4;
private javax.swing.JButton jButton5;

```

```

private javax.swing.JButton jButton6;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JPanel jPanel1;

}

AdminLogin.java:
import java.awt.HeadlessException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import javax.swing.JDialog;
import javax.swing.JOptionPane;
public class AdminLogin extends javax.swing.JFrame {

    /**
     * Creates new form AdminLogin
     */
    public AdminLogin() {
        initComponents();
        setLocationRelativeTo(null);
    }
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();

```

```

jLabel4 = new javax.swing.JLabel();
user = new javax.swing.JTextField();
pass = new javax.swing.JPasswordField();
jButton1 = new javax.swing.JButton();
jButton3 = new javax.swing.JButton();
jLabel1 = new javax.swing.JLabel();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

jPanel1.setLayout(null);

jLabel2.setFont(new java.awt.Font("Times New Roman", 1, 48)); // NOI18N
jLabel2.setForeground(new java.awt.Color(204, 255, 204));
jLabel2.setText("Admin Login");
jPanel1.add(jLabel2);
jLabel2.setBounds(130, 10, 290, 60);

jLabel3.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jLabel3.setForeground(new java.awt.Color(51, 51, 51));
jLabel3.setText("Username");
jPanel1.add(jLabel3);
jLabel3.setBounds(80, 150, 120, 50);

jLabel4.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jLabel4.setForeground(new java.awt.Color(51, 51, 51));
jLabel4.setText("Password");
jPanel1.add(jLabel4);
jLabel4.setBounds(80, 200, 120, 50);
jPanel1.add(user);
user.setBounds(230, 160, 190, 30);
jPanel1.add(pass);
pass.setBounds(230, 210, 190, 30);

```

```

jButton1.setBackground(new java.awt.Color(255, 255, 255));
jButton1.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jButton1.setText("Login");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});
jPanel1.add(jButton1);
jButton1.setBounds(130, 310, 110, 30);

jButton3.setBackground(new java.awt.Color(255, 255, 255));
jButton3.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jButton3.setText("Exit");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});
jPanel1.add(jButton3);
jButton3.setBounds(280, 310, 110, 30);

jLabel1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/book-
with-green-board-background_1150-3837.jpg"))); // NOI18N
jLabel1.setText("jLabel1");
jPanel1.add(jLabel1);
jLabel1.setBounds(0, -10, 538, 438);

javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);

```

```

        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addContainerGap()
                    .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, 539,
Short.MAX_VALUE)
                    .addContainerGap())
                );
        layout.setVerticalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addContainerGap()
                    .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, 419,
Short.MAX_VALUE)
                    .addContainerGap())
                );

        pack();
    } // </editor-fold>

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con
        =DriverManager.getConnection("jdbc:mysql://localhost:3306/Student","root","root");
        String sql = "SELECT * FROM  admin WHERE Username = ? AND Password = ?";
        PreparedStatement pst = con.prepareStatement(sql);
        pst.setString(1, user.getText());
        pst.setString(2, pass.getText());
        ResultSet rs = pst.executeQuery();
        if(rs.next()){
            JOptionPane.showMessageDialog(null, "Login Successfully....");
            setVisible(false);
        }
    }
}

```

```

        Admin a = new Admin();
        a.setVisible(true);
    }
    else{
        JOptionPane.showMessageDialog(null, "Login Failed....");
    }
}

catch(HeadlessException | ClassNotFoundException | SQLException e){
    JOptionPane.showMessageDialog(null, e);
}
}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    JDialog.setDefaultLookAndFeelDecorated(true);
    int response = JOptionPane.showConfirmDialog(null,"Do you want to EXIT?",
    "Confirm" ,
        JOptionPane.YES_NO_OPTION, JOptionPane.QUESTION_MESSAGE);
    if(response == JOptionPane.YES_NO_OPTION){
        setVisible(false);
        UserLogin ul = new UserLogin();
        ul.setVisible(true);
    }
}

public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
        javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    }
    catch (ClassNotFoundException ex) {
        java.util.logging.Logger.getLogger(Main.class).log(java.util.logging.
Level.SEVERE, "The class {0} is not found.", ex);
    }
    catch (InstantiationException ex) {
        java.util.logging.Logger.getLogger(Main.class).log(java.util.logging.
Level.SEVERE, "Could not instantiate the class {0}.", ex);
    }
    catch (IllegalAccessException ex) {
        java.util.logging.Logger.getLogger(Main.class).log(java.util.logging.
Level.SEVERE, "Could not access the class {0}.", ex);
    }
    finally {
        new Main().setVisible(true);
    }
}

```

```

        }
    }
} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(AdminLogin.class.getName()).log(java.util.logging
.Level.SEVERE, null, ex);

    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(AdminLogin.class.getName()).log(java.util.logging
.Level.SEVERE, null, ex);

    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(AdminLogin.class.getName()).log(java.util.logging
.Level.SEVERE, null, ex);

    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(AdminLogin.class.getName()).log(java.util.logging
.Level.SEVERE, null, ex);

    }
}
//</editor-fold>

/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        new AdminLogin().setVisible(true);
    }
});
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton3;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;

```

```

private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JPanel jPanel1;
private javax.swing.JPasswordField pass;
private javax.swing.JTextField user;
// End of variables declaration
}

```

StudLogin.java:

```

import java.awt.HeadlessException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import javax.swing.JDialog;
import javax.swing.JOptionPane;

public class StudLogin extends javax.swing.JFrame {

    /**
     * Creates new form StudLogin
     */
    public StudLogin() {
        initComponents();
        setLocationRelativeTo(null);
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();

```

```

jLabel2 = new javax.swing.JLabel();
jLabel3 = new javax.swing.JLabel();
user = new javax.swing.JTextField();
pass = new javax.swing.JPasswordField();
jButton1 = new javax.swing.JButton();
jButton2 = new javax.swing.JButton();
jButton3 = new javax.swing.JButton();
jButton4 = new javax.swing.JButton();
jLabel4 = new javax.swing.JLabel();
jLabel1 = new javax.swing.JLabel();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

jPanel1.setBackground(new java.awt.Color(204, 204, 204));
jPanel1.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0), 4));

jLabel2.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jLabel2.setText("Username");

jLabel3.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jLabel3.setText("Password");

user.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
user.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));

pass.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
pass.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));

jButton1.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jButton1.setText("Login");

```

```

        jButton1.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        });

        jButton2.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
        jButton2.setText("Reset");
        jButton2.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));
        jButton2.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton2ActionPerformed(evt);
            }
        });

        jButton3.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
        jButton3.setText("Exit");
        jButton3.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));
        jButton3.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton3ActionPerformed(evt);
            }
        });

        jButton4.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
        jButton4.setText("Sign-Up");
        jButton4.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));
        jButton4.addActionListener(new java.awt.event.ActionListener() {

```

```

        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton4ActionPerformed(evt);
        }
    });

    jLabel4.setIcon(new javax.swing.ImageIcon("C:\\Users\\Ashu\\Desktop\\Logo
for Project\\scholar-clipart-scholar-clipart-1.jpg.png")); // NOI18N

    javax.swing.GroupLayout jPanel1Layout = new
    javax.swing.GroupLayout(jPanel1);
    jPanel1.setLayout(jPanel1Layout);
    jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(42, 42, 42)
            .addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED_SIZE,
418, javax.swing.GroupLayout.PREFERRED_SIZE)

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(42, 42, 42)
                .addComponent(jButton1,
javax.swing.GroupLayout.PREFERRED_SIZE, 98,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(38, 38, 38)

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(38, 38, 38)
                .addComponent(jButton2,
javax.swing.GroupLayout.PREFERRED_SIZE, 115,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(38, 38, 38)

```

```

        .addComponent(jButton3,
javax.swing.GroupLayout.PREFERRED_SIZE, 110,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addComponent(jButton4,
javax.swing.GroupLayout.PREFERRED_SIZE, 115,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGroup(jPanel1Layout.createSequentialGroup())
        .addGap(18, 18, 18)

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)

        .addComponent(jLabel2)
        .addComponent(jLabel3))
        .addGap(105, 105, 105)

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING, false)

        .addComponent(user, javax.swing.GroupLayout.DEFAULT_SIZE,
237, Short.MAX_VALUE)
        .addComponent(pass))))
        .addContainerGap(121, Short.MAX_VALUE))
    );
    jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup())

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)

        .addGroup(jPanel1Layout.createSequentialGroup())
        .addGap(114, 114, 114)

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)

        .addComponent(jLabel2)

```

```

        .addComponent(user,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGap(61, 61, 61)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)

        .addComponent(jLabel3)

        .addComponent(pass,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGap(72, 72, 72)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)

        .addComponent(jButton1)

        .addComponent(jButton2)

        .addComponent(jButton3))

        .addGap(37, 37, 37)

        .addComponent(jButton4))

.addGroup(jPanel1Layout.createSequentialGroup())

        .addGap(78, 78, 78)

        .addComponent(jLabel4,
javax.swing.GroupLayout.PREFERRED_SIZE, 338,
javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addContainerGap(87, Short.MAX_VALUE))

);

jLabel1.setFont(new java.awt.Font("Times New Roman", 1, 48)); // NOI18N
jLabel1.setText("Student Login");

javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());

```

```

getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
            .addComponent(jLabel1)
            .addGap(376, 376, 376))
    );
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
            .addGap(27, 27, 27)
            .addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );

pack();
}

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con
=DriverManager.getConnection("jdbc:mysql://localhost:3306/Student","root","root");
        String sql = "SELECT * FROM signupdate WHERE Username = ? AND
Password = ?";
        PreparedStatement pst = con.prepareStatement(sql);
        pst.setString(1, user.getText());
        pst.setString(2, pass.getText());
        ResultSet rs = pst.executeQuery();
        if(rs.next()){
            JOptionPane.showMessageDialog(null, "Login Successfully....");
            setVisible(false);
            Student ma = new Student();
            ma.setVisible(true);
        }
        else{
            JOptionPane.showMessageDialog(null, "Login Failed....");
        }
    }
    catch(HeadlessException | ClassNotFoundException | SQLException e){
        JOptionPane.showMessageDialog(null, e);
    }
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    user.setText(null);
    pass.setText(null);
}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

```

```

        JDialog.setDefaultLookAndFeelDecorated(true);
        int response = JOptionPane.showConfirmDialog(null,"Do you want to EXIT?",
"Confirm" ,
        JOptionPane.YES_NO_OPTION, JOptionPane.QUESTION_MESSAGE);
        if(response == JOptionPane.YES_NO_OPTION){
            setVisible(false);
            UserLogin ul = new UserLogin();
            ul.setVisible(true);
        }
    }

    private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
        setVisible(false);
        NewUser nu =new NewUser();
        nu.setVisible(true);
    }

    public static void main(String args[]) {
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(StudLogin.class.getName()).log(java.util.logging.L
evel.SEVERE, null, ex);

        } catch (InstantiationException ex) {

```

```
java.util.logging.Logger.getLogger(StudLogin.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
```

```
    } catch (IllegalAccessException ex) {
```

```
java.util.logging.Logger.getLogger(StudLogin.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
```

```
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
```

```
java.util.logging.Logger.getLogger(StudLogin.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
```

```
    }
```

```
//</editor-fold>
```

```
java.awt.EventQueue.invokeLater(new Runnable() {
```

```
    public void run() {
```

```
        new StudLogin().setVisible(true);
```

```
    }
```

```
});
```

```
}
```

```
private javax.swing.JButton jButton1;
```

```
private javax.swing.JButton jButton2;
```

```
private javax.swing.JButton jButton3;
```

```
private javax.swing.JButton jButton4;
```

```
private javax.swing.JLabel jLabel1;
```

```
private javax.swing.JLabel jLabel2;
```

```
private javax.swing.JLabel jLabel3;
```

```
private javax.swing.JLabel jLabel4;
```

```
private javax.swing.JPanel jPanel1;
```

```
private javax.swing.JPasswordField pass;
```

```
private javax.swing.JTextField user;
```

```
}
```

Student.java:

```
import javax.swing.JDialog;
```

```

import javax.swing.JOptionPane;

public class Student extends javax.swing.JFrame {

    /**
     * Creates new form Student
     */
    public Student() {
        initComponents();
        setLocationRelativeTo(null);
    }
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jLabel2 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();
        jButton4 = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jPanel1.setLayout(null);

        jLabel2.setFont(new java.awt.Font("Times New Roman", 1, 48)); // NOI18N
        jLabel2.setForeground(new java.awt.Color(255, 255, 255));
        jLabel2.setText("Student Login");
        jPanel1.add(jLabel2);
        jLabel2.setBounds(40, 20, 310, 50);
    }
}

```

```

jButton1.setBackground(new java.awt.Color(0, 0, 0));
jButton1.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jButton1.setForeground(new java.awt.Color(255, 255, 255));
jButton1.setText("Add Info");
jButton1.setBorder(new javax.swing.border.LineBorder(new java.awt.Color(0, 0, 0), 1, true));
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});
jPanel1.add(jButton1);
jButton1.setBounds(130, 180, 210, 60);

```

```

jButton2.setBackground(new java.awt.Color(0, 0, 0));
jButton2.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jButton2.setForeground(new java.awt.Color(255, 255, 255));
jButton2.setText("View Info");
jButton2.setBorder(new javax.swing.border.LineBorder(new java.awt.Color(0, 0, 0), 1, true));
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});
jPanel1.add(jButton2);
jButton2.setBounds(130, 260, 210, 60);

```

```

jButton3.setBackground(new java.awt.Color(0, 0, 0));
jButton3.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jButton3.setForeground(new java.awt.Color(255, 255, 255));
jButton3.setText("View Marksheet");

```

```

        jButton3.setBorder(new javax.swing.border.LineBorder(new java.awt.Color(0, 0, 0), 1, true));
        jButton3.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton3ActionPerformed(evt);
            }
        });
        jPanel1.add(jButton3);
        jButton3.setBounds(130, 340, 210, 60);

        jButton4.setBackground(new java.awt.Color(0, 0, 0));
        jButton4.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
        jButton4.setForeground(new java.awt.Color(255, 255, 255));
        jButton4.setText("Logout");
        jButton4.setBorder(new javax.swing.border.LineBorder(new java.awt.Color(0, 0, 0), 1, true));
        jButton4.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton4ActionPerformed(evt);
            }
        });
        jPanel1.add(jButton4);
        jButton4.setBounds(810, 10, 100, 30);

        jLabel1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/certified-background-student-login-2.jpg"))); // NOI18N
        jLabel1.setText("jLabel1");
        jPanel1.add(jLabel1);
        jLabel1.setBounds(0, 0, 920, 500);

        javax.swing.GroupLayout layout = new
        javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);

```

```

        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addContainerGap()
                    .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, 922,
Short.MAX_VALUE)
                    .addContainerGap())
                );
        layout.setVerticalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addContainerGap()
                    .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, 500,
Short.MAX_VALUE)
                    .addContainerGap())
                );

        pack();
    } // </editor-fold>

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    setVisible(false);
    StudentData ul = new StudentData();
    ul.setVisible(true);
}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    JDialog.setDefaultLookAndFeelDecorated(true);
    int response = JOptionPane.showConfirmDialog(null,"Do you want to EXIT?",
"Confirm" ,
        JOptionPane.YES_NO_OPTION, JOptionPane.QUESTION_MESSAGE);
}

```

```

        if(response == JOptionPane.YES_NO_OPTION){
            setVisible(false);
            UserLogin ul = new UserLogin();
            ul.setVisible(true);
        }
    }

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
        setVisible(false);
        ViewInfo ul = new ViewInfo ();
        ul.setVisible(true);
    }

    private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look
and feel.
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

```

```
java.util.logging.Logger.getLogger(Student.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
```

```
    } catch (InstantiationException ex) {
```

```
java.util.logging.Logger.getLogger(Student.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
```

```
    } catch (IllegalAccessException ex) {
```

```
java.util.logging.Logger.getLogger(Student.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
```

```
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
```

```
java.util.logging.Logger.getLogger(Student.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
```

```
    }
```

```
//</editor-fold>
```

```
/* Create and display the form */
```

```
java.awt.EventQueue.invokeLater(new Runnable() {
```

```
    public void run() {
```

```
        new Student().setVisible(true);
```

```
    }
```

```
});
```

```
}
```

```
// Variables declaration - do not modify
```

```
private javax.swing.JButton jButton1;
```

```
private javax.swing.JButton jButton2;
```

```
private javax.swing.JButton jButton3;
```

```
private javax.swing.JButton jButton4;
```

```
private javax.swing.JLabel jLabel1;
```

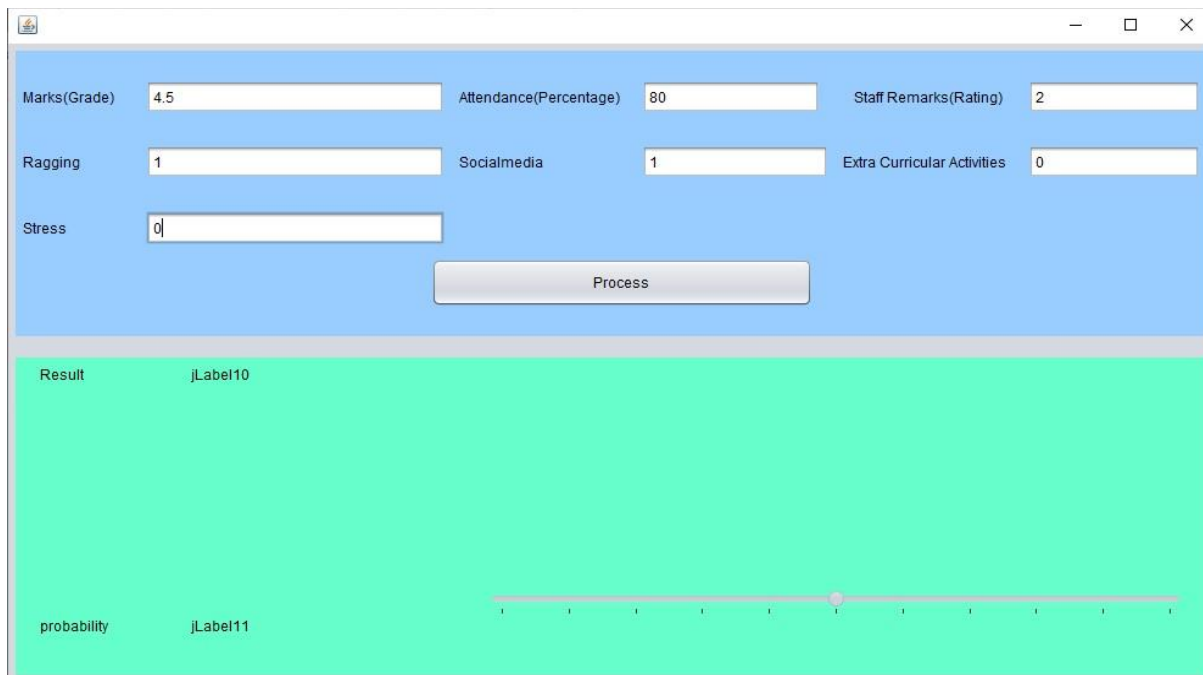
```
private javax.swing.JLabel jLabel2;
```

```
private javax.swing.JPanel jPanel1;
```

```
// End of variables declaration
```

```
}
```

B. SAMPLE SCREENSHOTS



A screenshot of a data entry application window. The window has a light blue header and a light green body. The header contains six input fields for data entry: Marks(Grade) with value 4.5, Attendance(Percentage) with value 80, Staff Remarks(Rating) with value 2, Ragging with value 1, Socialmedia with value 1, and Extra Curricular Activities with value 0. Below these fields is a 'Process' button. The body of the window is a large green area with a 'Result' label and a 'probability' label. A horizontal slider bar is positioned at the bottom of the green area, with a red dot indicating a value of approximately 0.5.

FIG B.1: Data Entry Page

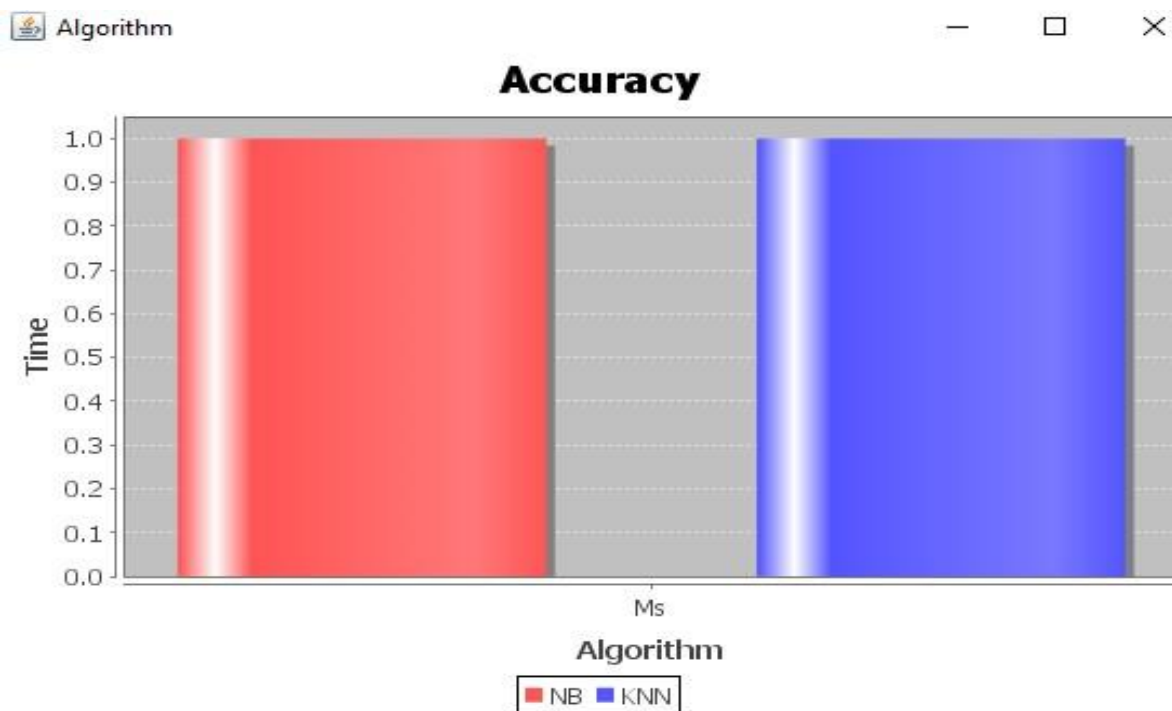


FIG B.2: Accuracy Diagram

C. RESEARCH PAPER

UNDERSTANDING STUDENTS PERFORMANCE USING MACHINE LEARNING TECHNIQUES

Akula Datta Veera Sasank Mitra
Department of Computer Science
& Technology
Sathyabama Institute of Science
and Technology
Chennai, India
sasankmitra04@gmail.com

Abhinay Sarveswara Rao Grandhi
Department of Computer Science
& Technology
Sathyabama Institute of Science
and Technology
Chennai, India
abhinayabhi123456789@gmail.com

Dr.R.Sethuraman
Department of Computer Science
& Technology
Sathyabama Institute of Science
and Technology
Chennai, India
sethuraman.cse@sathyabama.ac.in

ABSTRACT—

In many organizations, data mining techniques are used for analyzing large amount of available data's, information's for decision making process. In educational sector, Data mining is used for wide variety of application's such as performance of the students like mark, attendance, staff opinion, extracurricular activities, Ragging and stress. The data mining techniques used for identifying the performance of the student using K-means and KNN algorithms. These two algorithm's identifies and analyses the performance of the student. Subsequently, data preprocessing takes center stage, entailing data cleansing and feature engineering. Missing data is systematically addressed, outliers are carefully managed, and the dataset is meticulously prepared for in-depth analysis. Upon data preparation, the dataset is partitioned into training and testing sets, ensuring a robust evaluation of the machine learning algorithms' performance on unseen data, a vital step in establishing their generalization capabilities. The results of this study promise to provide actionable insights to enhance student performance, influencing strategies for personalized support, targeted resources, and pedagogical adjustments. Ethical considerations, including data privacy and fairness, are pivotal aspects of this research, reaffirming the commitment to responsible *and* equitable educational advancement.

I.INTRODUCTION

There are many tremendous improvement research interests in using machine learning in educational sector. This modern emerging sector, called educational data mining, concerned with improved methods that extract knowledge from data come from the educational sector. machine learning is a technique of sorting which is actually used to extract hidden patterns from huge databases. This concepts and methods can be applied in various fields like marketing, medicine, real estate, customer relationship, management, engineering, web mining, etc. Educational machine learning is a new emerging or advanced technique

personal information or academic performance.

Furthermore it can be achieve from elearning database systems which have a huge amount of data and information used by most institutes. It uses many techniques for proper implementation of machine learning concepts such as K-means Clustering and K-Nearest neighbour. Using these techniques different kinds of knowledge can be discovered using association rules, classification and clustering. By using this we extract knowledge that describes students performance in examination and all their detail information. From This huge amounts of data, the first task is to sort them out, cluster

analysis is to classify the raw data in a reasonable way.

Clustering is a bunch of physical or abstract objects, as per the degree of similarity between them, divided into several groups, and makes the same data objects within a groups of high similarity and different groups of data objects which are not similar.

Education is at the core of human development, and the quest to enhance educational outcomes is an ongoing endeavor. In this quest, the advent of data-driven technologies and machine learning has brought forth a paradigm shift in understanding and improving student performance. The presentation "Understanding Student Performance using Machine Learning Techniques" embarks on a journey to explore the profound impact of data-driven approaches in the educational landscape.

Traditionally, education relied on conventional teaching methods and limited data analysis to assess student performance. Educators depended on standardized tests, assignments, and subjective assessments to gauge how well students were absorbing and applying knowledge. While these methods have been invaluable, they are often retrospective, one-size-fits-all, and lack the capacity to provide a real-time, holistic view of students' academic be achieve from elearning database systems which have a journeys.

The Data Revolution:

Enter the data revolution. In this era, educational institutions

of machine learning that can be applied on the data related to educational institutes. The data of students can be academic records, assessment results, attendance records, online learning activities, and surveys. This data deluge has introduced new possibilities and challenges. It is in this context that data-driven approaches are gaining prominence. The rise of data-driven education is underpinned by a desire to better understand student behaviors and academic performance. It seeks to tailor education to the unique needs of each student, to detect early warning signs of academic struggles, and to predict and mitigate learning obstacles. It is an approach that emphasizes personalized learning, timely interventions, and data-informed decision making, with the ultimate goal of enhancing student success.

Central to this data-driven transformation are machine learning techniques and data analysis. These technologies are adept at sifting through vast datasets, identifying patterns, and making predictions. They enable educational institutions to gain unprecedented insights into student behavior and performance.

The objectives of this presentation are threefold:

1. Exploration: We will embark on an exploratory journey through the landscape of data-driven education. We will examine the sources of data, delve into the methodologies used for data analysis, and investigate the machine learning algorithms leveraged for understanding student performance.

2. Insight: With each step of this exploration, we aim to unearth actionable insights that empower educators, institutions, and policymakers to make data-informed decisions. These insights can aid in the identification of at risk students, the personalization of learning experiences, and the continuous improvement of education.

3. Transformation: Ultimately, this presentation seeks to illuminate how the fusion of data-driven approaches, machine learning, and education has the potential to transform the educational landscape. By harnessing the power of data, we can create a future where every student's educational journey is tailored, supported, and enriched by data insights.

As we embark on this educational odyssey, we invite you to join us in understanding the profound ways in which data-driven approaches, machine learning techniques, and a commitment to student success are reshaping the educational landscape. Together, we explore how education is not just evolving; it is undergoing a data driven revolution that has the potential to empower students and educators in unprecedented ways.

II.LITERATURE SURVEY

1. Prediction of Students Academic Performance using Data Mining: Analysis

Author/Year: Sayana T S/ 2018

have gained access to a wealth of data. This data encompasses a wide spectrum of information, including clustering algorithm and Decision tree

Results: In this paper, there is a evaluation between students association rule algorithm, K-means clustering algorithm and Decision tree. This survey for assessing students performance is based on different attributes. Attributes include class quizzes, mid and final exam marks, assignment, lab works etc. We discuss the procedure based on Decision Tree of data mining method, Data Clustering that enable academicians to predict students performance and instructor can take essential step to develop student academic performance.

Advantages: The efficiency of various decision tree algorithms can be analyzed based on their accuracy and time taken to derive the tree. The predictions obtained from the system have helped the tutor to identify the weak students and improve their Performance. The analysis of the result declared from the university is a proof for the same. Since the application of data mining brings a lot of advantages in higher learning institution.

Disadvantages: That is the number of cluster, must be determined earlier. It is difficult to determine the number of clusters when there is slight change in the data.

2.Predicting Student Performance Using Data Mining and Learning Analytics Techniques: A Systematic Literature Review

Author/Year: Abdallah Namoun and Abdullah Alshankiti/ 2020

Techniques: Artificial Neural Networks, Support Vector Machine, K-Nearest Neighbor, and Random Forests.

Results: The best practices for conducting systematic literature reviews, e.g., PICO and PRISMA, were applied to synthesize and report the main results. The attainment of learning outcomes was measured mainly as performance class standings (i.e., ranks) and achievement scores (i.e., grades). Regression and supervised machine learning models were frequently employed to classify student performance. Finally, student online learning activities, term assessment grades, and student academic emotions were the most evident predictors of learning outcomes. We conclude the survey by highlighting some major research challenges and suggesting a summary of significant recommendations to motivate future works in this field.

Advantages: Identified that attributes and methods that predict student performance Applied the PICO methodology

Disadvantages: These reviews suffered from several limitations, for they (1) were generally broad, (2) did not focus on using student outcomes as an indicator of student performance, (3) suffered from quality issues (e.g.methodologies not thoroughly defined), and (68.7% for passed and 88.8% for failed). (4) were not published in highly indexed venues

Techniques: Association rule algorithm, K-means

3.Students Performance Prediction Using Data Mining

Techniques

Author/Year: Urgesh Ugale, Jeet Pawar, Sachin Yadav, Dr. Chandrashekhar Raut/ 2020

Techniques: Decision Tree, Naive Bayes, Support Vector Machine, K-Nearest Neighbors

Results: In this approach, preprocessing step will be applied to raw dataset so that the mining algorithm will be applied properly. The prediction about students' performance can help him/her to enhance the performance..

Advantages: Did not address student outcome from different perspectives – Did not discuss the limitations – Did not compare the predictive models

Disadvantages: Formulated well defined questions

4.Student Performance Prediction Model based on Supervised Machine Learning Algorithms

Author/Year: Ali Salah Hashim, Wid Akeel Awadh, Alaa Khalaf Hamoud/ 2020

Techniques: Decision Tree, Naïve Bayes, Logistic Regression, Support Vector Machine, K-Nearest Neighbour

Results: In this study, we compared the performances of several supervised machine learning algorithms. Results indicated that logistic regression classifier is the most accurate in predicting the exact final grades of students (68.7% for passed and 88.8% for failed).

Advantages: Listed the techniques used in the learning analytics for predicting student performance

Disadvantages: Did not assess the quality of the studies – Did not compare the model

5.Students Performance Prediction Using Data Mining Techniques

Author/Year: Durgesh Ugale, Jeet Pawar, Sachin Yadav, Dr. Chandrashekhar Raut/ 2020

Techniques: Decision Tree, Naive Bayes, Support Vector Machine, K-Nearest Neighbors

Results: In this approach, preprocessing step will be applied to raw dataset so that the mining algorithm will be applied properly. The prediction about students' performance can help him/her to enhance the performance.

Advantages: Did not address student outcome from different perspectives – Did not discuss the limitations – Did not compare the predictive models

Disadvantages: Formulated well defined questions

6.Student Performance Prediction Model based on Supervised Machine Learning Algorithms

Techniques: Decision Tree, Naïve Bayes, Logistic Regression, Support Vector Machine, K-Nearest Neighbour

Results: In this study, we compared the performances of several supervised machine learning algorithms. Results

Disadvantages: Presented a preliminary study lacking crucial details .Did not publish the full results of the survey

indicated that logistic regression classifier is the most accurate in predicting the exact final grades of students (68.7% for passed and 88.8% for failed).

Advantages;Listed the techniques used in the learning analytics for predicting student performance

Disadvantages: Did not assess the quality of the studies – Did not compare the model

7. Prediction of Students Academic Performance using Data Mining: Analysis

Author/Year: Sayana T S/ 2018

Techniques: Association rule algorithm, K-means clustering algorithm and Decision tree

Results: In this paper, there is a evaluation between students association rule algorithm, K-means clustering algorithm and Decision tree. This survey for assessing students performance is based on different attributes. Attributes include class quizzes, mid and final exam marks, assignment, lab works etc. We discuss the procedure based on Decision Tree of data mining method, Data Clustering that enable academicians to predict students performance and instructor can take essential step to develop student academic performance.

Advantages: The efficiency of various decision tree algorithms can be analyzed based on their accuracy and time taken to derive the tree. The predictions obtained from the system have helped the tutor to identify the weak students and improve their Performance. The analysis of the result declared from the university is a proof for the same. Since the application of data mining brings a lot of advantages in higher learning institution.

Disadvantages: That is the number of cluster, must be determined earlier. It is difficult to determine the number of clusters when there is slight change in the data.

8. An Exploratory Study on Data Mining in Education: Practiced Algorithms and Methods

Author/Year: Ancelmo Castro, Leandro Garcia, David Prata, Marcelo Lisboa, and Monica Prata/ 2017

Techniques: K-means, Decision Tree, Classification and Regression Tree, J48

Results:Results showed that different algorithms and methods are used in data mining for education. Many of these algorithms are essential for classification and data control on a large scale. The algorithms C5.0, C4.5, and K means, were highlighted methods in this study. For 48% of the articles, these three mining algorithms are most commonly used to analyze data in large scale, especially in education. Decision tree was the main method applied, with 45% of the algorithms applied, probably because of the graphic knowledge representation, which could help experts to better interpret the elicited evidences, and to postulate causes and effects.

Advantages: Reported the overall context of student outcomes

9. Predicting Student Academic Performance using Support Vector Machine and Random Forest

Author/Year: Leena H. Alamri , Ranim S. Almuslim, Mona S. Alotibi, Dana K. Alkadi, Irfan Ullah Khan, Nida Aslam/ 2020

Techniques: Support Vector Machines (SVM) and Random Forest (RF). Binary classification and regression techniques have been applied with both SVM and RF.

Results: The experimental results for both SVM and RF algorithms applied to both datasets showed that the accuracy in the case of binary classification achieves a superior accurate prediction reach to 93%, while in regression, the lowest RMSE is 1.13 in case of RF.

Advantages: It is easy and fast to predict class of test data set. It also perform well in multi class prediction

Disadvantages: SVM is also known as a bad estimator, so the outputs of probability from predict are not to be taken too important.

10. A Comparative Study to Predict Student's Performance Using Educational Data Mining Techniques

Author/Year: Annisa Uswatun Khasanah, Harwati / 2017

Techniques: Decision Tree, Neural Network, Naïve Bayes, K-Nearest neighbour.

Results: The outcome showed that student's attendance and GPA in the first semester were in the top rank from all Feature Selection methods, and Bayesian Network is outperforming Decision Tree since it has higher accuracy rate.

Advantages: It perform good in situations of categorical input

variables compared to numerical variable(s).

Disadvantages: Data may be over-fitted or over-classified if small sample is tested.

III. METHODOLOGY

The work aims to develop a trust model using data mining techniques, which mines required information, so that the present education system may adopt this as a strategic management tool.

The proposed system use educational data mining techniques

to evaluate performance and identify undesirable behavior. In educational sector, Data mining is used for wide variety of application's such as performance of the students like mark, attendance, staff opinion, extracurricular activities, Ragging and stress. The data mining techniques used for identifying the performance of the student using K-means and KNN algorithms.

Educational database contain the useful information for Evaluating Students.

on weather condition. Let's follow the below steps to perform it.

Step 1: Convert the data set into a frequency table

The data mining techniques are more helpful in classifying educational database and help us in evaluating the performance and undesirable behavior of a student.

As of now, Existing system take only performance into consideration which is not sufficient for having system, which can help us to evaluate performance of a student.

We are not having a system which would help us to integrate the performance and undesirable into consideration.

Existing system miss the undesirable data for the students. And It may not check the social data for the student

PROPOSED ALGORITHM

K-NEAREST NEIGHBOR(KNN) CLASSIFICATION METHOD

K-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The k-NN algorithm is among the simplest of all machine learning algorithms. The neighbors are taken from a set of objects for which the class (for k-NN classification) or the object property value (for k-NN regression) is known.

STEP 1: BEGIN STEP 2: Input: $D = \{(x_1, c_1), \dots, (x_N, c_N)\}$

STEP 3: $x = (x_1, \dots, x_n)$ new instance to be classified

STEP 4: FOR each labelled instance (x_i, c_i) calculate $d(x_i, x)$

STEP 5: Order $d(x_i, x)$ from lowest to highest, $(i = 1, \dots, N)$

STEP 6: Select the K nearest instances to x: D_{kx}

STEP 7: Assign to x the most frequent class in D_{kx}

STEP 8: END

K-MEANS CLUSTERING ALGORITHM

What is K-means?

1. Partitional clustering approach

2. Each cluster is associated with a centroid (center point)

3. Each point is assigned to the cluster with the closest centroid

4 Number of clusters K must be specified

Details of K-means

1. Initial centroids are often chosen randomly. - Clusters produced vary from one run to another 2. The centroid is (typically) the mean of the points in the cluster.

3. 'Closeness' is measured by Euclidean distance, cosine similarity, correlation, etc.

4. K-means will converge for common similarity measures mentioned above.

5. Most of the convergence happens in the first few iterations. - Often the stopping condition is changed to 'Until relatively few points change clusters'

NAIVE BAYES ALGORITHM

Let's understand it using an example. Below I have a training data set of weather and corresponding target variable 'Play' (suggesting possibilities of playing).

Step 2: Create Likelihood table by finding the probabilities like Overcast probability = 0.29 and probability of playing is 0.64.

Step 3: Now, use Naive Bayesian equation to calculate the posterior probability for each class. The class with the highest posterior probability is the outcome of prediction.

Data Collection:

This phase is pivotal as it serves as the foundation upon which the entire project rests. In the realm of education, data collection encompasses the systematic gathering of information from a multitude of sources. These sources include but are not limited to academic records, demographic data, attendance records, online learning activities, and surveys. Each of these sources contributes unique insights into students' behaviors, interactions, and, most importantly, their academic performance.

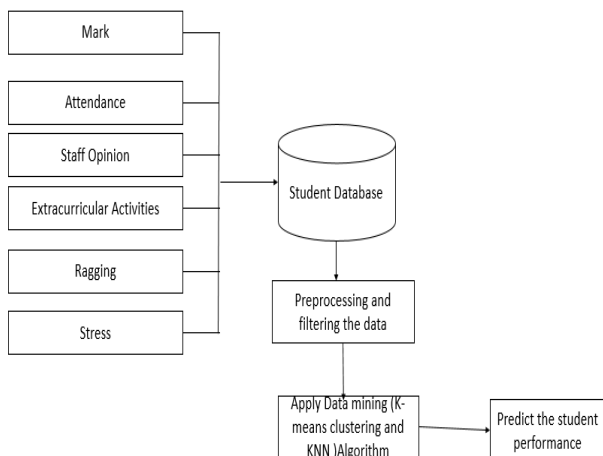
Data Preprocessing:

Data preprocessing, following data collection, is a critical step in the project's journey. It is the process of cleaning and preparing the collected data to ensure its quality, consistency, and suitability for machine learning analysis. The integrity of the data is paramount, as it directly influences the quality and reliability of the machine learning models that will be developed.

This stage involves several key tasks. First, it includes handling missing values in the dataset. Missing data can lead to inaccurate analysis, strategies for imputation or dealing with missing data must be employed. outliers in the data, which are data points that significantly deviate from the norm, must be identified to avoid skewing the analysis.

In summary, data preprocessing is a critical step in the data analysis pipeline. It guarantees that the data is well structured, consistent, and free from anomalies, setting the stage for the development of robust machine learning models that can ultimately contribute to the project's overarching goal of enhancing education and improving student success.

ARCHITECTURE



educational outcomes, promote student success, and create.

IV. OBJECTIVES

The main aim of this project is to improve the student performance in studies based on some important factors. Education is an essential element for the betterment and progress of a country. It enables the people of a country civilized and well mannered. Now-a-days developing new methods to discover knowledge from educational database in order to analyse student

's trends and behaviours towards

education. To analyse the data from different dimensions categorize it and to summarize the relationships. It motivated us to work on student dataset analysis. The data collection, categorization and classification is being performed manually. It uses many techniques for proper implementation of machine learning concepts such as K means Clustering and K-Nearest neighbour. Using these techniques different kinds of knowledge can be discovered using association rules, classification and clustering. By using this we extract knowledge that describes students performance in examination and all their detail information.

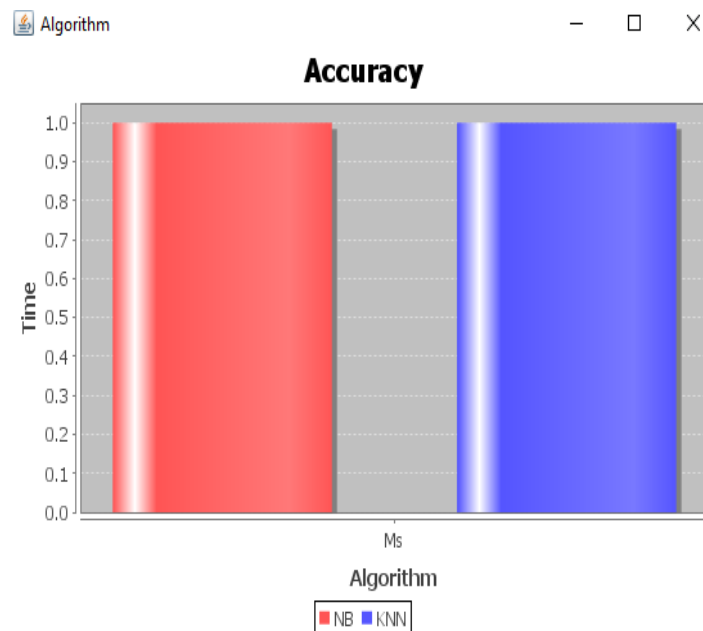
V. CONCLUSION

In the pursuit of enhancing educational outcomes and student success, the convergence of data-driven methodologies and machine learning techniques represents a transformative force. Our project, which sought to explore and leverage the power of these approaches, has illuminated the potential to revolutionize the educational landscape. Through comprehensive data collection and meticulous

data preprocessing, we unearthed invaluable insights into student behavior and academic performance. Machine learning models, adept at identifying patterns and making predictions, offered a deeper understanding of the complex relationship between various factors and student outcomes. This project has underscored the significance of data informed decision-making in education. By leveraging predictive models, educators and institutions gain the ability to detect early warning signs of academic struggles, personalize learning experiences, and continuously improve the education provided to students.

As we conclude this project, it is clear that we stand at the threshold of a new era in education. An era where each student's academic journey is enriched by data-driven insights, enabling timely interventions and personalized support. It is a vision where education is not merely evolving; it is undergoing a data-driven revolution that has the potential to empower students and educators in unprecedented ways.

In this rapidly changing educational landscape, it is imperative to embrace and harness the power of data and machine learning for the betterment of education. We invite all stakeholders - educators, institutions, policymakers, and researchers - to join us in this journey to enhance



VI .REFERENCES

[1] M.Durairaj, C.Vijitha .Educational Data mining for Prediction of Student performance Using Clustering Algorithms. The data mining techniques are more helpful inclassifying educational database Which contain the useful information for predicting a student's performance.

[2] Kin Fun Li, David Rusk and Fred Song.Predicting Student Academic Performance,The performance predictors,if identified, can then be used effectively to formulate corrective action plans to improve the attrition.

[3] E.Osmanbegovic, Mirza Suljic.Data Mining Approach For Predicting Student Performance,Using data mining the aim was to develop a model which can derive the conclusion on student's academic success.

[4] A. Banumathi,A. Pethalakshmi. A novel approach for upgrading Indian education by using data mining techniques.

[5] Anoopkumar M ; A. M. J. Md. Zubair Rahman,A Review on Data Mining techniques and factors used in Educational Data Mining to predict student amelioration.

[6] K. Vanlehn, “The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems,” *Educ. Psychol.*, vol. 46, pp. 197–221, Oct. 2011

[7] A. Janson, M. Sollner, and J. M. Leimeister, “Ladders for learning: Is scaffolding the key to teaching problem-solving in technology mediated learning contexts,” *Acad. Manage. Learn. Educ.*, vol. 19, no. 4, 439–468, 2020.

[8] M. C. Kim and M. J. Hannafin, “Scaffolding problem solving in technology-enhanced learning environments (TELEs): Bridging research and theory with practice,” *Comput. Educ.*, vol. 56, 403–417, Feb. 2011.

[9] T. A. Brush and J. W. Saye, “A summary of research exploring hard and soft scaffolding for teachers and students using a multimedia supported learning environment,” *J. Interact. Learn.*, vol. 1, pp. 1–12, Oct. 2002.

[10] R. Knote, A. Janson, and J. M. Leimeister, “Value co-creation in smart services: A functional affordances perspective on smart personal assistants,” *J. Assoc. Inf. Syst.*, vol. 4, pp. 418–458, Mar. 2020

D. CERTIFICATE

