Informatics in Education, 2024, Vol. 23, No. 1, 1–24

DOI: 10.15388/infedu.2024.04

Enhancing Student Performance Prediction via Educational Data Mining on Academic Data

Zareen ALAMGIR[1*], Habiba AKRAM[1], Saira KARIM[1], Aamir WALI[2]
[1]*Department of Computer Science, National University of Computer and Emerging Sciences Pakistan*
[2]*Department of Data Science, National University of Computer and Emerging Sciences Pakistan*
e-mail: zareen.alamgir@nu.edu.pk, l181848@lhr.nu.edu.pk, saira.karim@nu.edu.pk, aamir.wali@nu.edu.pk

Received: March 2023

Abstract. Educational data mining is widely deployed to extract valuable information and patterns from academic data. This research explores new features that can help predict the future performance of undergraduate students and identify at-risk students early on. It answers some crucial and intuitive questions that are not addressed by previous studies. Most of the existing research is conducted on data from 2–3 years in an absolute grading scheme. We examined the effects of historical academic data of 15 years on predictive modelling. Additionally, we explore the performance of undergraduate students in a relative grading scheme and examine the effects of grades in core courses and initial semesters on future performances. As a pilot study, we analyzed the academic performance of Computer Science university students. Many exciting discoveries were made; the duration and size of the historical data play a significant role in predicting future performance, mainly due to changes in curriculum, faculty, society, and evolving trends. Furthermore, predicting grades in advanced courses based on initial pre-requisite courses is challenging in a relative grading scheme, as students' performance depends not only on their efforts but also on their peers. In short, educational data mining can come to the rescue by uncovering valuable insights from academic data to predict future performances and identify the critical areas that need significant improvement.

Keywords: educational data mining, student performance prediction, machine learning, computer science, learning analytics.

1. Introduction

A stable higher education system plays a vital role in the advancement and growth of a nation. Higher education is essential for progress as it can equip individuals with mod-

* Corresponding author

ern knowledge, skills, and technology. Unfortunately, being able to retain students in universities is becoming a major challenge (Veloso *et al.*, 2023; Tight, 2020). Particularly in developing countries, universities strive to build a powerful student force for numerous domains and fields. With limited resources and various financial and social constraints of students in developing countries, it is cumbersome for academic institutions to retain students and equip them with the essential skill set (B.K. Fomba, 2023).

Computer science (CS) is a fast-evolving domain that needs strong foundations. Universities aim to make students exceptional computer scientists that can perform well in different areas of computer science. Compromise in the CS core courses is therefore not an option, and there is a need to adopt some strategies to identify the students at risk (Erika B. Varga, 2021). Current and historical student data can be utilized to extract interesting patterns regarding student progress. Surveys can be conducted to acquire student information about their behavior and concerns. The institutions can use this information and develop new approaches to facilitate students in different ways (Hoffait and Schyns, 2017). They can organize events or sessions to target students who need assistance. An institute can be successful and produce outstanding graduates only if it knows the issues faced by its students and resolve them.

One of the biggest challenges educational institutions face today is learning from data and extracting valuable information. Each institute is unique as it follows different criteria and rules based on its region and norms. Hence, they are independent, and uniform methods cannot be applied to discover useful patterns from data effortlessly (Abu Saa *et al.*, 2019). This generates problems regarding the best way to capture, organize, and productively use the data. An area of research that has emerged to address student data is known as educational data mining (EDM). This field is responsible for creating, researching, and using computerized methods to find hidden patterns in massive data sets, which may be challenging to analyze due to large data volumes. Recently, the analysis of academic data, such as learning analytics, academic data mining, predictive analytics, and student analytics, has emerged as a new area of research. The similarity between all these principles is the use of educational data. The institutions are interested in understanding student academic performance. However, this is a difficult task, and a large number of factors such as economic, social, demographic, cultural, and educational background can impact learning outcomes (Batool *et al.*, 2023).

The contribution of this research work is multi-fold. It addresses compelling questions regarding undergraduate studies in the computer science (CS) domain. Not much work has been done in EDM that focuses on mining the academic data of undergraduate students enrolled in the CS degree program that follows a relative grading scheme. In relative grading, the student's grades depend not only on their performance but also on the performance of their peers. The previous studies work on the data of a few years (at most 5); however, we have gathered and analyzed the data of 15 years. The historical data give more samples to properly train the machine learning (ML) models and overcome the issue of overfitting. Furthermore, historical data can better adapt the ML model to changes in faculty, curriculum, teaching methodology, and students' attitudes and behaviors. Besides this, in a technical field like CS, the advanced level CS courses are

based on the concepts taught in foundation (pre-requisite) courses. This study attempts to utilize the students' performance in pre-requisite courses to forecast student performance in advanced-level courses.

The research questions adopted in this study are novel and, to the best of our knowledge, have not been dealt with before. Hence, this research adds value to the literature in many ways and tries to fill the gap left by previous studies. The research questions addressed by this study are as follows:

1. Which attributes can help predict student performance in the CS domain that follows a relative grading scheme?
2. Is the historical data more helpful in predicting student performance, or does the chunk of last few years give better prediction results?
3. Can we predict student performance in advanced CS courses based on pre-requisite CS courses?
4. In the specific context of the study, which ML model, among Random forest, Neural network and Linear regression, performs best?

The paper is structured as follows. Section 2 presents a detailed literature review. Section 3 introduces the data and discusses the methodology, and Section 4 is the experiment Section that highlights the findings. Lastly, Section 5 gives the conclusion and ideas for future work.

2. Literature Review

Academic performance prediction is one of the most important applications of educational data mining and learning analytics. Student performance prediction is a broad term that includes various aspects and types of performance prediction, including course-based performance, yearly performance, and graduating grade prediction. This section conducts a comprehensive review of the latest research done in educational data mining for the academic performance prediction of undergraduate students based on different factors and characteristics. The main factors that are considered include performance prediction based on pre-admission data, prediction based on academic data of initial years in an undergraduate program, and effects of using low-cost and non-academic variables for performance prediction. Table 1 categorizes the research studies on factors that are crucial for performance prediction.

Some studies utilized pre-admission data to predict student performance and dropouts in different educational fields (Tan *et al.*, 2022). Adekitan and N-Osaghae (2019) used scores of various standardized tests required for seeking university admission to predict the performance of first-year students. An accuracy of only around 50% was obtained, thus indicating a very weak correlation between the admission requirements and first-year performance. Martínez-Navarro *et al.* (2021) used the university's admission time data that includes demographic features like the city of residence to mine important features that can predict poor performance and dropout. Alharthi (2021) utilized the university's pre-admission data to predict the performance of students in the health sciences program. The attributes include high school grade point average (GPA), general aptitude

Table 1
Categorization of studies based on factors used for students' performance prediction

Factors for Categorizing Research Studies	Input Features and Performance Focus	Studies
Performance Prediction using pre-admission data	High school GPA, GAT and Admission test scores (predict performance at the end of first year of undergrad program)	(Tan *et al.*, 2022), (Martínez-Navarro *et al.*, 2021), (Alharthi, 2021), (Erika B. Varga, 2021), (Adekitan and N-Osaghae, 2019)
Performance Prediction using academic data of undergrad program's initial years	GPA of first–second year of undergrad program and grades in a some courses (predict graduating GPA)	(Hashim *et al.*, 2020), (Qazdar *et al.*, 2019), (Miguéis *et al.*, 2018), (Asif *et al.*, 2017), (Hoffait and Schyns, 2017), (Jia and Maloney, 2015)
Performance Prediction using low-cost variables	Class participation, resource availability, heterogeneity, and class strength (predict future academic performance)	(Tomasevic *et al.*, 2020), (Yousafzai *et al.*, 2020), (Xu *et al.*, 2019), (Helal *et al.*, 2018), (Sandoval *et al.*, 2018), (Thiele *et al.*, 2016), (Xing *et al.*, 2015)
Performance Prediction using non-academic variables in addition to academic data	Behavioral and emotional characteristics, social and demographic features (forecast future academic performance)	(Wild *et al.*, 2023), (Kukkar *et al.*, 2023), (Yao *et al.*, 2019), (Nti *et al.*, 2022), (Karagiannopoulou *et al.*, 2021), (Keser and Aghalarova, 2021), (Fernandes *et al.*, 2019), (Thiele *et al.*, 2016)
Performance Prediction in Courses	Marks in course assessments: assignments, quizzes, home work and midterm (predict course grades)	(Wang *et al.*, 2023), (Mai *et al.*, 2022), (Yağcı, 2022), (Injadat *et al.*, 2020b), (Injadat *et al.*, 2020a), (Ahmad *et al.*, 2015), (Bydžovská, 2016), (Marbouti *et al.*, 2016), (Costa *et al.*, 2017)

test (GAT) scores, and a few others. Various machine learning techniques were used, and the best results were obtained using a Random forest.

Academic attributes and student performance in the initial years can be good indicators to predict future performance and identify at-risk students. Miguéis *et al.* (2018) proposed a model to forecast the overall academic performance of undergraduate students based on the data collected at the end of the first academic year. The model deploys different classification techniques: Random forest, Decision trees, Naïve Bayes, and Support vector machines. The authors suggested that the prediction can be improved by considering course performance and student activities. Asif *et al.* (2017)

predicted students' performance at the end of the undergraduate degree program based on the first and the second year's results using various classifiers. This study allowed teachers and program directors to overcome performance issues by focusing on relevant students. Hashim *et al.* (2020) also conducted a study on bachelor students to find students at high risk of dropping out before the final examination and provide them extra care and guidance. Qazdar *et al.* (2019) employed data mining techniques on the first semester results of 10 subjects (in the physics stream) to predict students' performance in the national exam for the Bac (baccalaureate) certificate. Some researchers focused on the undergraduate students struggling in the first year of admission (Hoffait and Schyns, 2017; Jia and Maloney, 2015) and predicted the performance for the second year. In (Hoffait and Schyns, 2017), authors considered students from different degree programs and conducted a 'What-if' Sensitivity analysis to determine if the student performance could be improved by changing or optimizing a few features.

Few researchers attempt to identify different and novel factors that can affect students' performance (Tomasevic *et al.*, 2020; Thiele *et al.*, 2016). Xing *et al.* (2015) introduced a new perspective for performance prediction based on student participation in studies. They develop a comparatively different model that uses interpretable genetic programming. It is observed that student participation in a course is a key factor in determining performance. In another research (Yousafzai *et al.*, 2020), a genetic algorithm is used to select 29 optimal features to predict performance in the exams. Sandoval *et al.* (2018) proposed a prediction model based on low-cost variables to identify the undergraduate students struggling due to large student strength enrolled for a course. Helal *et al.* (2018) considered student heterogeneity while constructing models for predicting students' academic performance. In addition to this, data from an online learning management system was included to analyze the effects of student engagement with online learning. In another study, Xu *et al.* (2019) focused on the facilities provided to students as an indicator of student performance prediction. They believe that facilities depict student behavior; hence they explored internet usage behavior to estimate students' seriousness in their studies.

According to some studies, only academic factors are not enough for performance prediction, but socio-demographic factors can be beneficial too (Thiele *et al.*, 2016; Wild *et al.*, 2023; Kukkar *et al.*, 2023; Nti *et al.*, 2022). Yao *et al.* (2019) predicted the performance of undergraduate students based on their behavior in school. They considered three factors: diligence, orderliness, and sleep. Keser and Aghalarova (2021) use non-academic attributes such as demographic, social, emotional, parents' job, and alcohol consumption to predict academic performance in two schools. An accuracy of above 90% indicates that non-academic variables strongly correlate with the student's academic performance. However, the dataset is relatively small and spans over one academic year. Karagiannopoulou *et al.* (2021) use students' emotional characteristics along with the pace of study to predict academic progress. Fernandes *et al.* (2019) highlighted student factors that affect performance. They used a gradient boosting machine to predict the students' performance at the end of the school year. The result showed that demographic attributes like students' school, age, and neighborhood also play an important role in addition to academic features.

Some researchers (Marbouti *et al.*, 2016) focused on early performance prediction in courses using internal variables related to courses that are available to instructors. The performance of students in early course assignments, quizzes, and midterm exams can also help identify troubled students. Wang *et al.* (2023) employed an advanced XGBoost SAP module to forecast course performance. Few studies explored the performance of undergraduate students in the computer science (CS) domain (Ahmad *et al.*, 2015; Yağcı, 2022). However, almost all employed a small sample size and few features. Bydžovská (2016) predicted the final grade of a course based on two approaches: regression and collaborative filtering. Injadat *et al.* (2020b, 2020a) develop a multi-split bagging ensemble model that uses the Gini index and p-value to predict the student's performance during the course rather than at the end of the course. This work is helpful in the early identification of problematic students. Costa *et al.* (2017) observed that failures in the introductory courses in undergrad studies have a worse effect on an overall student's academic performance. They conducted experiments on two datasets collected from introductory programming courses and applied four prediction techniques, and among them, the Support vector machine provided good results.

The existing studies have various shortcomings. Some are sample biased and deal with only one learning school or few courses. Most deal with a small dataset and do not utilize historical data. Moreover, the few studies that use a large dataset do not consider the performance in important courses. This work adopted a novel and comprehensive approach that explores undergraduate computer science student performance from different angles and in multiple scenarios. We include features related to student performance in foundation courses and previous semesters. Furthermore, we also examine the role of the historical data of 15 years for predicting student performance. Our proposed method can give useful feedback to instructors regarding students' performance so they can implement appropriate models. This study aims to help students by promoting quality education and reducing the number of failures

3. Methods and Data

This study analyzes the academic performance of undergraduate students in the CS domain from different perspectives and attempts to answer some crucial research questions that are not addressed by previous studies. The aim is to extract useful information from the data, get insight, predict future performance, and identify the critical areas that need significant improvement. For this purpose, we use predictive modelling, a process that employs advanced data mining and machine learning techniques to predict the outcomes. Predictive modelling can be beneficial in predicting student academic performance and providing an understanding of trends that may happen in the near future. It is widely used in academics, research, and development.

Our basic approach is illustrated in Fig. 1. First, the student data is pre-processed; this involves handling missing values and cleaning the data. In cleaning step, academic attributes such as initial semesters' GPAs (grade point averages), batch and grades

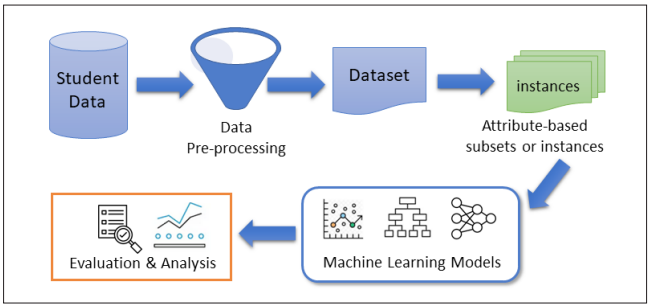

Fig. 1. The block diagram for the proposed model.

of CS courses are retained. Next, feature subset selection techniques are applied to identify the best features for the task at hand. The selected attributes are checked for correlation to remove redundant attributes. After the data has been extensively pre-processed, it is ready for analysis. Different instances and subsets are extracted from the pre-processed dataset to perform various experiments and find answers to proposed research questions.

We employed three machine learning models for analysis: linear regression, random forest, and neural networks. These models have been carefully selected, considering factors such as the type and size of the input data and ensuring a wide range of modelling approaches are covered to determine the best machine learning model for the problem at hand. Furthermore, different evaluation metrics are used to thoroughly analyze and comprehend the outcomes of both regression and classification models, enabling us to gain valuable insights from the results. For regression models, commonly used evaluation metrics such as root mean squared error (RMSE) and mean absolute error (MAE) are employed. RMSE measure the average squared difference between predicted and actual values, while MAE calculates the average absolute difference between predicted and actual values, providing a measure of the average prediction error. For classification models, evaluation metrics such as F1 score and area under the receiver operating characteristic curve (AUC) are used. The Fscore provides a balance between precision and recall, while AUC evaluates the model's performance across different classification thresholds. By employing these evaluation metrics, we aim to analyze the performance of the regression and classification models for predicting students' future performance.

To ensure the credibility and generalizability of our research, it is crucial to acknowledge and address potential threats to the validity of our research. While our study leverages ML models to predict student performance, several key factors can influence the reliability of the results. First, the quality of data used for training and testing the ML models can impact the predictive performance and may introduce noise or bias, affecting the accuracy of the predictions. Even the ML models can potentially threaten external validity as their performance relies on the algorithm's assumptions, input features, hyperparameter tuning, and specific implementation details. Furthermore, the choice of predictors, such as previous semester GPA (grade point average),

also introduces potential limitations. While GPA serves as a widely used indicator of academic performance, it may not capture the full range of factors contributing to a student's future success or adequately account for external influences beyond academic performance.

To mitigate these threats to validity, we have taken numerous precautions. We have employed rigorous data pre-processing techniques, such as handling missing values and addressing data quality issues. Additionally, we have carefully selected and implemented a range of ML models, considering their strengths and limitations, and have performed extensive model evaluation and validation procedures. We have experimented with different input features to find the most reliable predictors for forecasting student performance. While our research strives to provide valuable insights into predicting student performance using previous semester GPA, acknowledging and addressing these threats to validity is essential for a comprehensive interpretation of our results and future research endeavors.

3.1. *Data*

The data is obtained from a renowned local university. It consists of the transcripts of the graduated students from the CS department from 2001–2015. The research performed on educational data mainly involves the data that belongs to a particular educational institute. The reason is to analyze the trends and discover meaningful information hidden in data that can be useful for the institution. The students' characteristics and environment vary from place to place, so we cannot apply or suggest the same rules or techniques for every institution around the region. Thus, there is always a need to discover new characteristics amongst the students of diverse universities and discover new trends by applying different techniques.

The dataset consists of academic attributes and a few demographic attributes. It includes the details regarding the courses taken by the students, and the grades and GPA (grade point average) scored. The number of courses an undergraduate CS student takes in the four-year degree program is around 40. We focus mainly on core computer science courses that are pre-requisite for advanced CS courses and are essential for building solid foundations in the computer science domain. The intuition is that poor performance in pre-requisite courses may identify students facing issues in essential logic development and coding. This can lead to the early identification of the students who need extra help.

The seven important CS courses offered during the four-year program are selected after careful scrutinization. The courses included are Introduction to Computing (ITC), Computer Programming (CP), Data Structures (DS), Database Systems (DB), Object-Oriented Analysis and Design (OOAD), Design and Analysis of Algorithms (ALGO), and Software Engineering (SE). The courses like ITC and CP are introductory CS courses offered in the first two semesters, while others are advanced. The introductory courses are pre-requisite for advanced courses. The selected courses are considered the core CS courses by the Higher Education Commission (HEC).

The original dataset before pre-processing has 3687 instances. It is multivariate and consists of nominal and numeric attributes. The attributes extracted from the transcript are as follows:

- **Batch**: This feature indicates the year of the enrolment of the student. It is an important feature as the dataset spans over fifteen years from 2001 to 2015, and during this period, there could be significant changes in the course contents, teaching methodology, and faculty. Hence, the performance of the students may vary. We have included this feature to study the effects of time and fluctuation on student performance.
- **Semester GPAs**: The GPAs scored by the students in the first four semesters and the final cumulative GPA (CGPA) are included; this gives five numeric attributes.
- **Grades**: The grades of students for the selected courses: ITC, CP, DS, DB, OOAD, Algo, and SE are included. Hence, we have seven nominal attributes, one for the grade of each selected course. The grade of a course can take a value of A, B, C or D. The data consists of the transcript of students who have completed the degree, so the final grade of a course cannot be F. If a student gets an F, he repeats the course. The grades are encoded to numerical values 1–4.
- **Repeat counts**: Repeat count indicates the number of times a student repeated a particular course. The student has to repeat a course if he fails or wishes to increase the grade in a course. We have seven repeat count attributes, one for each selected CS course (ITC, CP, DS, DB, OOAD, Algo, and SE).

The key attributes and their description is given in the Table 2.

Table 2
Key Input Features and their Description

No	Feature	Description
1	Batch	Year of the enrollment of the student
2	ITC	Grade in the course Introduction to Computer Science (ITC)
3	CP	Grade in the course Computer Programming (CP)
4	DS	Grade in the course Data Structure (DS)
5	DB	Grade in the course Database Systems (DB)
6	OOAD	Grade in the course Object Oriented and Design (OOAD)
7	ALGO	Grade in the course Design and Analysis of Algorithms
8	SE	Grade in the course Software Engineering
9	Sem1	GPA scored by a student in the first semester
10	Sem2	GPA scored by a student in the second semester
11	Sem3	GPA scored by a student in the third semester
12	Sem4	GPA scored by a student in the fourth semester
13	CGPA	Graduating CGPA, that is, Cumulative GPA scored in BS(CS) program
14	RC	Repeat Count indicates the number of times a student has repeated a course. We have seven RC attributes one for each selected course (ITC, CP, DS, DB, OOAD, Algo and SE).

3.2. *Data Pre-processing and Exploration*

Before the data could be used, it was pre-processed. The data was first cleaned by removing irrelevant fields such as name, age, and others. The roll numbers of the students were also removed due to privacy concerns. We also handled missing values, outliers, and skewness. Furthermore, we also analyzed the attributes and calculated the correlation between them.

3.2.1. *Missing Values*

The data contained missing and null values. Some students have transferred from different campuses or fields, and their transcripts list only the courses accepted and transferred with no GPA information. All missing GPAs for courses were filled with the mean GPA of the student. Some students transferred to other campuses or left the program after one or two semesters. Hence, their records were removed as they could not be included in the final analysis. The pre-processed data has 2326 student records from the year 2001 to 2015.

3.2.2. *Attribute Correlation*

It would be interesting to see if a strong correlation exists between any attributes in the data. Fig. 2 shows a correlation heatmap of 13 key attributes of the dataset generated using the Pearson correlation coefficient given in equation 1.

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{1}$$

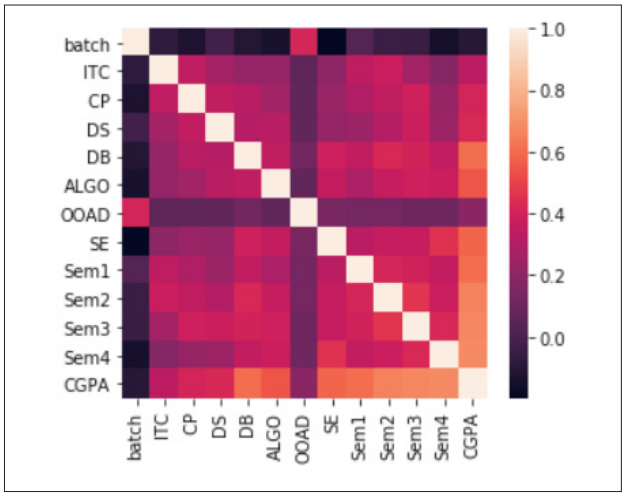

Fig. 2. Correlation heatmap of 13 key attributes
(including the dependent variable graduating CGPA).

It is evident from the Fig. 2 that there is no strong correlation between key attributes other than CGPA, which is a dependent attribute in this study. Hence, each selected attribute provides a unique perspective and can help predict the student's performance. Furthermore, the relatively high correlation of the final CGPA with the rest of the attributes signifies that these attributes are essential in predicting the graduating CGPA. However, it is interesting to note that Batch and OOAD GPA correlate little with other attributes, even with the CGPA (the dependent attribute). It would be exciting to study the effect of these attributes on the prediction of CGPA.

3.3. *Prediction and Classification*

We have employed different machine learning models to analyze student data and discover valuable insights. For regression analysis, we used: Linear regression (LR), Random forest regressor (RFR), and multilayer perceptron regressor (MLPR). These models have been carefully selected after thorough scrutiny, taking into account several factors to ensure the most appropriate choices. One crucial consideration was the type and size of the input data. Our dataset consists of tabular data rather than images, so deep learning models like convolutional neural networks (CNNs), primarily designed for image analysis, are unsuitable. To cover a wide range of modelling approaches, we have included models from different categories. Regression models, known for their ability to estimate relationships between variables, are chosen to capture linear or non-linear associations within the dataset. Tree-based models, such as decision trees and random forests, are selected for their interpretability and capacity to handle complex interactions and non-linear relationships in the data. Additionally, we also included a multilayer perceptron, which is a feed-forward neural network. Neural Networks are AI-based human-inspired models offering unique perspectives and techniques that complement traditional approaches. By including a diverse set of models, we aim to gain a comprehensive understanding of how each type performs on our specific educational dataset. This approach allows us to assess the strengths and limitations of different modelling techniques and determine which models are best suited for the specific educational dataset under consideration in this study.

For classification, we have utilized: Logistic Regression (LoR), random forest (RF), and a multilayer perceptron (MLP). An MLP classifier provided in the sklearn library is used. The network has two hidden layers. The activation function used in the hidden layers is the logistic function, and the identity function is used in the output layer. Conjugate gradient descent is applied as the optimization algorithm. We also explored Adam's optimization, but it did not make any significant difference. Similarly, we used sklearn RF, RFR, LR, and LoR implementations available in the sklearn library. For linear regression, the simple linear regression model is used.

We performed 10-fold cross-validation to achieve better results. In 10-fold cross-validation, the data is divided into ten blocks, where nine are used to train the model, and one is used for testing purposes. This process is repeated ten times, once for each block. In addition, parameters are tuned using RandomizedSearchCV from the sklearn library. In order to tune their parameter, hyperparameter tuning is applied.

3.4. *Evaluation Criteria and Measures*

The evaluation measures used for the validation of regression models are Mean absolute error (MAE) and Root mean square error (RMSE). These metrics are widely used for estimating the performance of the classification and regression models. MAE is the mean of the absolute differences between actual and predicted values, while RMSE is the standard deviation of the predicted errors. MAE and RMSE are calculated as follows:

$$MAE(x, \hat{x}) = \frac{\sum_{i=1}^{n} \mid x_i - \hat{x_i} \mid}{n} \tag{2}$$

$$RMSE(x, \hat{x}) = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \hat{x_i})^2}{n}} \tag{3}$$

Here x is the actual value, $\hat{x}$ is the predicted value, and n is the number of samples in the test set. Both evaluation metrics provide a good estimate of model performance. However, RMSE gives high weight to large errors as it squares the prediction errors before averaging.

For classification, we have employed AUC, an area under the ROC (receiver operating characteristic curve), and Fscore as evaluation metrics. A ROC curve is a graph that depicts the performance of a classification model at all classification thresholds, and AUC gives an aggregate measure of the model's performance across different classification thresholds, depicting the trade-off between true positive rate and false positive rate. The Fscore (F-measure) provides a balance between precision and recall. Precision quantifies the model's ability to correctly identify positive instances, while recall (also known as sensitivity or true positive rate) measures the model's ability to identify all positive instances correctly. The Fscore, the harmonic mean of the precision and recall, measures a test's accuracy in binary classification and is calculated as follows:

$$Fscore = \frac{2 * precision * recall}{precision * recall} \tag{4}$$

The values of AUC and Fscore range between 0–1, where a value closer to 1 is considered better. Generally, a value between 0.7 and 0.8 is good, while a value greater than 0.8 is considered excellent. By employing these evaluation metrics, we aim to assess the performance of the regression and classification models and gain deeper insights into their predictive capabilities, identify areas of improvement, and make informed decisions about their suitability for the task at hand.

4. Computational Experiments and Results

This research deployed advanced machine learning models to find answers to the research questions posed in this study (Section 1). The most important aspect that our study explores is the role of students' performance in early semesters and core CS courses in identifying the students facing difficulties in grasping CS concepts. First, we attempt to identify the attributes in data that can help predict the student's future performance. Second, we delve into the data to understand the effect of historical data in identifying challenging students; more concretely, we wish to determine whether an institution's historical data is more helpful in predicting grades than the last few years' data. Third, we use the grades in pre-requisite courses to foresee students' performance in advanced CS courses in a relative grading scheme. This study can help us determine the effect of changing teaching methodologies, policies, and faculty on students' grades.

We have carefully designed and performed rigorous experiments on different subsets of our data to find answers to the above intrigues and discover valuable insights from collected data.

4.1. *Experiment 1: Identify Useful Features in Data for Predicting Student Performance*

In this section, we carried out a series of experiments to find attributes that help estimate student performance in the undergraduate program in Computer Science. One aspect of analyzing student performance is to predict the final graduating CGPA (cumulative grade point average) based on GPAs scored in initial semesters. The problem under study is a regression problem, so we used three widely used regression algorithms: linear regression (LR), random forest regressor (RFR), and multilayer perceptron regressor (MLPR). In this experiment, we used the pre-processed data of 2326 student records from 2001 to 2015.

4.1.1. *Experiment 1a: Examine the Role of GPA in Initial Semesters for Predicting the Graduating CGPA*

We extracted students' GPAs from the data for the first four semesters and attempted to predict the final CGPA using regression algorithms. Table 3 shows the results of the experiments conducted to estimate the graduating CGPA using the different combinations of initial semester GPAs. Fig. 3 shows the RMSE for different regression models in a horizontal bar chart; the smaller values of the error measure indicate a better fit. The regressor that performed best in almost all cases is RFR. The best result of RMSE = 0.179 and MAE = 0.136 is achieved by RFR when the first four semester GPAs are given as input. The error measures try to capture the difference between the predicted and actual values, and as errors are very close to zero, this indicates that the model is giving a reasonable estimate of the student graduating CGPA based on the results of the initial semesters.

Table 3
Predicting students' graduating CGPA using GPAs scored in initial semesters (Sem)

Methods	LR		RFR		MLPR	
Input Features	MAE	RMSE	MAE	RMSE	MAE	RMSE
Sem 1	0.274	0.343	0.264	0.331	0.304	0.380
Sem 1–2	0.221	0.280	0.227	0.29	0.245	0.307
Sem 1–3	0.189	0.238	0.183	0.231	0.217	0.273
Sem 1–4	0.147	0.191	0.136	0.179	0.165	0.213

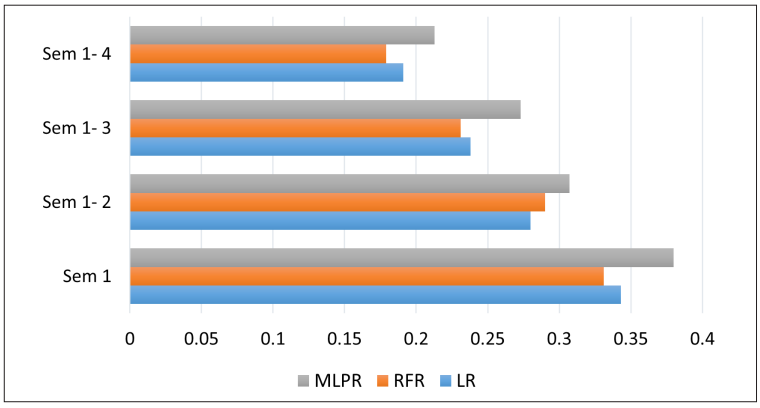

Fig. 3. RMSE of different regression models for predicting students' graduating CGPA using initial semesters' GPAs.

Furthermore, the correlation coefficient in the experiment was 0.89, indicating that the model captures around 90% of the data variance. The error values decrease when more initial semester GPAs are included as input attributes. This indicates that to get a reasonable estimate of future student performance and find problematic students; we need GPAs of at least 3–4 semesters.

4.1.2. *Experiment 1b: Examine the Benefit of Additional Features for Predicting the Graduating CGPA*

The previous experiment showed that GPA scores by students in the initial semesters could help predict their future performance at the end of their program. In this experiment, we explore the effect of additional features in improving student performance prediction. We consider additional features such as grades scored in core CS courses, repeat counts for each course, and the batch of students. The repeat count indicates the number of attempts a student made before passing the course.

First, we used the grades scored by students in core CS courses in addition to the GPAs of initial semesters. We wish to examine whether students' performance in core CS courses helps forecast graduating CGPA. The core CS courses that are considered include ITC, CP, DS, DB, ALGO, OOAD, and SE. Table 4 shows that the prediction

improves as we add the grades scored by the student in different core courses. We added courses one by one in the order they are offered to students in various semesters during their curriculum. The error decreases as more courses are included as an input variable. Hence, we can conclude that each course plays a role in improving the prediction of CGPA. The best result with MAE = 0.113 and RMSE = 0.147 was achieved by RFR when grades of all seven courses were included, in addition to initial semester GPAs. The error is very close to zero, indicating that predicted grades are very close to the actual ones. Fig. 4 shows the RMSE of different regression models in a horizontal bar graph. It is evident from the graph that RFR is producing the minimum error, and error decreases significantly as we add grades of different courses.

Next, we evaluate the benefit of features like batch and repeat count (RC) of different courses in predicting students' performance. The results show that the error de-

Table 4

Predicting CGPA using grades in CS courses and CGPAs in initial semesters

Methods	LR		RFR		MLPR	
Additional Input Features	MAE	RMSE	MAE	RMSE	MAE	RMSE
ITC	0.147	0.191	0.136	0.178	0.164	0.212
ITC, CP	0.146	0.190	0.135	0.177	0.161	0.209
ITC, CP, DS	0.143	0.186	0.133	0.174	0.162	0.210
ITC, CP, DS, DB	0.131	0.171	0.125	0.163	0.160	0.206
ITC, CP, DS, DB, OOAD	0.126	0.166	0.122	0.159	0.156	0.204
ITC, CP, DS, DB, OOAD, ALGO	0.120	0.159	0.118	0.155	0.155	0.203
ITC, CP, DS, DB, OOAD, ALGO, SE	0.114	0.150	0.113	0.147	0.142	0.186

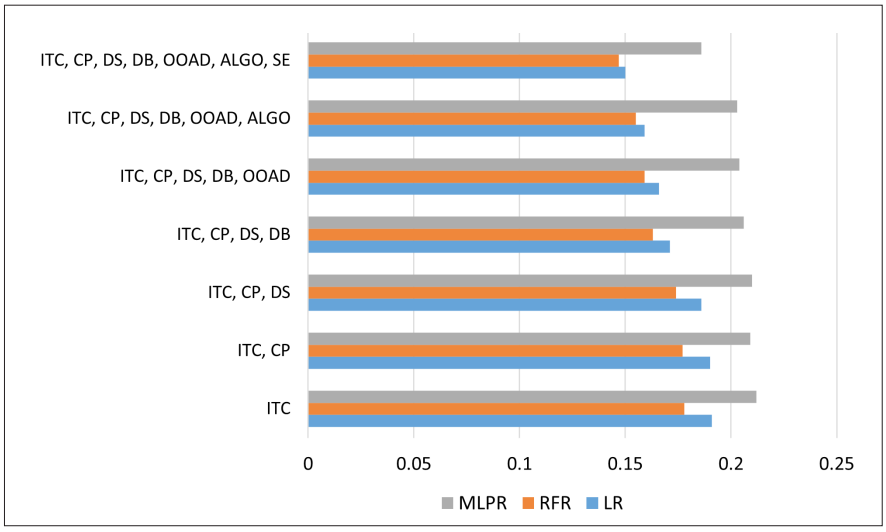

Fig. 4. RMSE of regression models for predicting CGPA using initial semesters' GPAs and grades in CS courses.

creases with the inclusion of repeat count features. The repeat count feature indicates the number of times a student repeats a particular course. Besides getting an F grade, a student can also repeat a course to improve the previous passing grade.

Table 5 shows the MAE and RMSE for different combinations of the input features, while Fig. 5 shows the RMSE for different regression models. It is evident from the figure that GPAs scored by students in the first four semesters are crucial indicators for predicting student performance. The grades scored by students in core courses help improve performance prediction if used in addition to semester GPAs. However, if they are used without semester GPAs, then MAE and RMSE is quite huge; RF gives MAE = 0.205, RMSE = 0.258, while LR gives MAE = 0.192 and RMSE = 0.24. The repeat count feature further helps to reduce error. However, it is interesting to observe that the batch feature does not have any significant effect. This finding is in line with the attribute correlation heatmap, which showed that the batch feature does not have any correlation with CGPA. The classifier that performs best in most of scenarios is RFR. The performance of LR is comparable; however, MLPR does not perform well and produces relatively higher MAE in all experiments.

Table 5
Predicting students' graduating CGPA based on different combination of input features

Methods	LR		RFR		MLPR	
Input Features	MAE	RMSE	MAE	RMSE	MAE	RMSE
Courses' grades	0.192	0.240	0.205	0.258	0.225	0.284
Sem 1–4	0.147	0.191	0.136	0.179	0.165	0.213
Sem 1–4, Courses' grades	0.114	0.150	0.113	0.147	0.142	0.186
Sem 1–4, Courses' grades, RCs	0.112	0.149	0.111	0.145	0.139	0.188
Sem 1–4, Courses' grades, RCs, Batch	0.112	0.149	0.112	0.146	0.142	0.191

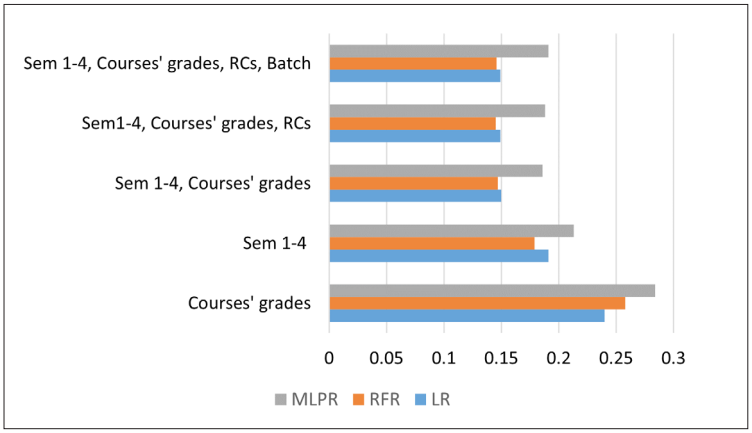

Fig. 5. RMSE of regression models for predicting CGPA using different combination of input features.

4.2. *Experiment 2: Examine the Effect of Data-size and Time-period in Predicting Students Performance*

We want to analyze the effect of data size and the time period in estimating student performance in the degree program. We have utilized data of 15 years in this experiment. During the 15 years, there could be significant changes in the curriculum, teaching methodologies, and students' attitudes. Through this experiment, we aim to determine whether an institution's historical data is more helpful in predicting students' performance than the last few years' data or vice versa.

We conducted various experiments using different subsets of data. First, we used data subsets consisting of records for five consecutive years, that is, 2001–2005, 2006–2010, and 2011–2015. The number of records in each data subset varies depending on the number of students enrolled each year. The feature set consists of GPAs of the first four semesters, grades in core courses, and repeat counts for core courses. The experiments are conducted using LR, RFR, and MLPR. The regressor that gives the minimum error is LR on the dataset for years (2011–2015). Next, we repeat the experiments with the dataset consisting of three consecutive years. The MAE and RMSE for predicting CGPA for various periods are shown in Table 6 along with the number of instances for each period. Fig. 6 shows the RMSE generated by different regression models for each time period. The error produced by MLPR is relatively high, while the results of RFR and LR are close and comparable.

It is clear from the results that more data means better results, especially in the case of advanced classifiers like RFR and MLPR. We compared the values over different time-period shown in Table 6, and it is evident that the value of MAE is low for historical data in almost all scenarios. Even though in 15 years, there may be many changes in curriculum, faculty, teaching methodologies, students characteristics, and course contents. The historical data provides ample samples to classifiers to learn the behavior and other dynamics necessary for identifying the problematic students. An exception occurred in the case of LR as it performed well with an MAE of 0.104 in the 5-year time period 2011–2015. The excellent performance of LR in the time period 2011–2015

Table 6
Predicting students' graduating CGPA based on data of different years

		Methods	LR		RFR		MLPR	
Time-period		Data-size	MAE	RMSE	MAE	RMSE	MAE	RMSE
1* 15-years	**2001–2015**	2326	0.112	0.149	**0.111**	**0.145**	0.139	0.188
3* 5-years	**2001–2005**	510	0.122	0.162	0.122	0.160	0.179	0.265
	2006–2010	765	0.117	0.154	0.121	0.158	0.149	0.218
	2011–2015	1051	0.104	0.138	0.109	0.139	0.133	0.181
5* 3-years	**2001–2003**	418	0.121	0.165	0.120	0.160	0.191	0.276
	2004–2006	247	0.113	0.142	0.124	0.153	0.170	0.227
	2007–2009	459	0.126	0.169	0.134	0.175	0.178	0.251
	2010–2012	516	0.113	0.139	0.115	0.142	0.151	0.207
	2013–2015	686	0.114	0.150	0.113	0.143	0.159	0.226

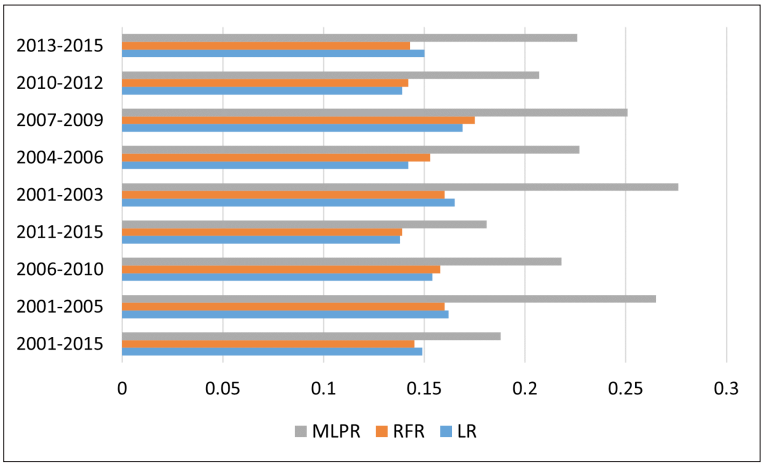

Fig. 6. RMSE of regression models for predicting CGPA using data of different years.

is mainly because of the reasonable number of data records (approximately 1051). The other reason could be fewer fluctuations in this period. However, this trend is not observed for other five-year datasets. For smaller datasets and periods, LR produced the best performance, and for more extended time periods, RFR was better able to capture the data dynamics. The MLPR did not perform well in any scenario mainly because it needed huge amounts of data to train and learn.

4.3. *Experiment 3: Predict Performance in Advanced CS Courses Based on Pre-requisite Courses*

This experiment's main objective is to predict student performance in an advanced-level CS course based on his performance in pre-requisite courses and previous semesters. The problem under study is a binary classification problem as we want to determine whether students would perform well in a course. We used the data of 15 years, from 2001 to 2015, to train machine learning models for this binary classification task.

The experiments were conducted on four advanced CS courses: DB, OOAD, ALGO, and SE. For each course, a separate subset or an instance of the dataset was generated. The number of records in each course instance/dataset is 2326, while the number of features can vary depending on the pre-requisite courses. The detail of the pre-requisite of each course is given in Table 7. For the DB dataset, we use the grade obtained by students in prerequisite courses: ITC, CP, and DS to predict the grade in DB. While in the case of ALGO and OOAD, the grades in ITC, CP, DS, and DB are considered. Lastly, for predicting the grade of SE, we used the grades in ITC, CP, DS, DB, and OOAD.

The experiments were carried out using three different classifiers: logistic regression (LoR), random forest (RF), and multilayer perceptron (MLP). Table 8 shows the area under the ROC curve (AUC) and Fscore values for DB, OOAD, ALGO, and SE. As this is a classification problem, evaluation metrics AUC and Fscore are used. AUC gives an

Table 7

Pre-requisite courses for different advanced Computer Science Courses

Course	Pre-requisite Courses
DB	ITC, CP, DS
ALGO	ITC, CP, DS, DB
OOAD	ITC, CP, DS, DB
SE	ITC, CP, DS, DB, OOAD

Table 8

Predicting performance in advanced CS courses based on grades in pre-requisite courses

Methods	LoR		RF		MLP	
Course	AUC	Fscore	AUC	Fscore	AUC	Fscore
DB	0.754	0.691	0.729	0.661	0.731	0.673
ALGO	0.748	0.702	0.728	0.678	0.712	0.677
OOAD	0.746	0.701	0.732	0.698	0.741	0.692
SE	0.774	0.698	0.755	0.689	0.747	0.692

aggregate measure of performance across different classification thresholds. Fscore, the harmonic mean of the precision and recall, is a measure of a test's accuracy in binary classification. The values of AUC and Fscore range between 0–1, where a value closer to 1 is considered better. Generally, a value between 0.7 to 0.8 is good, while a value greater than 0.8 is excellent.

Each CS course has different pre-requisite course(s) and difficulty levels. Therefore, the value of AUC varies for each course. Fig. 7 shows the vertical bar graph to depict the AUC values for various regression models, as this is a classification problem; hence we have used a vertical bar graph. It is evident from Fig. 7 that the value of AUC for predicting the student's performance based on historical data is reasonable. And the

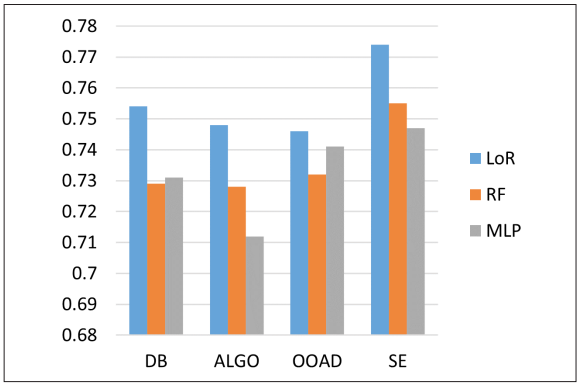

Fig. 7. AUC of classification models for predicting performance in advanced CS courses using grades in prerequisite courses.

best prediction is made for the SE dataset with values of AUC and Fscore as 0.774 and 0.698, respectively. The LoR performed significantly better than RF and MLP. The performance of MLP and RF are comparable for some courses.

4.4. *Discussion and Analysis*

This study investigates the performance of undergraduate students in the computer science domain from various dimensions and aspects. It attempts to resolve some crucial and intuitive queries not addressed by previous studies (Miguéis *et al.*, 2018; Xu *et al.*, 2017; Injadat *et al.*, 2020b). The main objective is to extract valuable information from the data, get insight, predict future performance, and identify the critical areas that need significant improvement.

Several academic, social, and behavioral factors can influence the student's performance. The early classification of undergraduate students by their academic potential can help identify students struggling with concepts and need support. Furthermore, this can allow the university to formulate strategies for mitigating failures, improving result performance, and adequately managing the institution's resources. The attributes related to behaviors are challenging to attain as students give false information in surveys. So this study relies primarily on academic attributes and attempts to determine how students perform and survive in a highly competitive environment where courses are technical and heavily based on the concepts learned in the previous courses.

In this work, we designed various experiments to explore and examine the different perspectives and discover valuable answers to the questions posed in the introduction section.

Question: Which attributes can help predict student performance in the CS domain that follows a relative grading scheme?

We observed that GPAs scored by students in the initial semesters are good indicators for predicting performance and identifying problematic students. The prediction power of the ML model can be significantly enhanced if we also include the grades of students in initial core CS cores along with the information on how many times students repeated the course to attain that grade. The data used in this study is based on a relative grading scheme in which student grades are awarded based on their performance compared to other students' performance. Thus, a student's course grade depends not only on his efforts but also on his peers' performance. We conclude from the various experiments that student performance can be predicted in a relative grading scheme. The current research work (Iqbal *et al.*, 2017; Sandoval *et al.*, 2018) mainly concentrated on the system that uses standards-based grading. No significant work exists to handle the system with relative grading, an emerging system used in many international tests.

Question: Is the historical data more beneficial in predicting student performance, or does a chunk of the last few years give better prediction results?

With years, there can be significant changes in faculty, curriculum, and student behavior. So we wish to determine if historical data improves or degrades the quality of perfor-

mance prediction. We conducted experiments using data of fifteen years from 2001–2015 to predict students' graduating CGPA, then we repeated the experiments with data of 5 consecutive years and also with 3 consecutive years. It is evident from the results that more data improves the ML model and enhances its predictive qualities. At least data of 3–5 years should be used to train the ML model. The existing studies (Miguéis *et al.*, 2018) used data of at most a year or two; hence they failed to predict the student performance correctly and capture the variation in faculty and curriculum.

Question: Can we predict student performance in advanced CS courses based on pre-requisite CS courses?

An advanced CS course builds on the concepts learned in pre-requisite courses. If a student lacks a stronghold on basic concepts, this can drastically degrade his performance in the advanced level course. Previously, researchers focused on predicting course-based student performance but did not include pre-requisite courses (Marbouti *et al.*, 2016). Furthermore, the current studies worked with standards-based grading and did not identify the impact of a relative grading scheme (Iqbal *et al.*, 2017). Different experiments show that we can predict a student's performance in an advanced-level CS course using his grades in pre-requisite courses and his GPA in the initial semesters. The prediction can be further improved if we include the initial assessment of the advanced course.

Question: In the specific context of the study, which classifier, among Random forest, Neural network, and Linear Regression, performs best?

The Random forest performs best, and the performance of Linear Regression is comparable. However, probably due to reasonably small data, the Neural network (MLP) does not show outstanding results in most experiments.

A comprehensive set of experiments were performed to predict the performance of the students in the computer science domain and to answer the research questions. This study has addressed all the research questions.

5. Conclusion and Future Work

Early detection of students struggling with concepts and logic development is crucial in the technical field, like computer science. This work digs deep into the data to discover features that can help identify the problematic students early on in the bachelor's degree in CS. Several academic, social, and behavioral factors can influence students' performance. The attributes related to behaviors are challenging to attain as students give false information in surveys. So this study relies primarily on academic attributes and attempts to determine how students perform and survive in a highly competitive environment where courses are technical and heavily based on the concepts learned in the previous courses.

Furthermore, we investigate the effect of historical data in predicting student performance in a system that follows a relative grading scheme. The relative grading is more challenging as the student's performance depends not only on his grades but also on the

grades of his peers. Finally, we attempt to predict student grades in advanced courses based on his performance in foundation courses offered in the initial years.

As future work, we can improve the prediction of students' grades in advanced CS courses by feeding initial assessments such as quizzes and assignments as input features. Furthermore, we can deploy data mining techniques such as clustering to segment the students early on and improve the prediction quality. Another direction of future work could be to assign weights to input features according to their importance using entropy method to enhance student performance prediction quality.

Funding

No Funding.

Compliance with Ethical Standards

This statement is to certify that the author list is correct. The Authors also confirm that this research has not been published previously and that it is not under consideration for publication elsewhere. On behalf of all Co-Authors, the Corresponding Author shall bear full responsibility for the submission. There is no conflict of interest. The student data has been obtained from a local university. The ID of the students were removed by the university to hide the identity of the student before the data is granted. Furthermore, the name of the university is kept anonymous for privacy and ethical reasons.

References

Abu Saa, A., Al-Emran, M., Shaalan, K. (2019). Factors affecting students' performance in higher education: a systematic review of predictive data mining techniques. *Technology, Knowledge and Learning*, 24, 567–598.

Adekitan, A.I., N-Osaghae, E. (2019). Data mining approach to predicting the performance of first year student in a university using the admission requirements. *Education and Information Technologies*, 24(2), 1527–1543.

Ahmad, F., Ismail, N.H., Aziz, A.A. (2015). The prediction of students' academic performance using classification data mining techniques. *Applied Mathematical Sciences*, 9(129), 6415–6426.

Alharthi, H. (2021). Machine Learning Techniques to Predict Academic Performance of Health Sciences Students. In: *2021 20th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*, pp. 33–36. IEEE.

Asif, R., Merceron, A., Ali, S.A., Haider, N.G. (2017). Analyzing undergraduate students' performance using educational data mining. *Computers & Education*, 113, 177–194.

Batool, S., Rashid, J., Nisar, M.W., Kim, J., Kwon, H.-Y., Hussain, A. (2023). Educational data mining to predict students' academic performance: A survey study. *Education and Information Technologies*, 28(1), 905–971.

B.K. Fomba, P.N. D.F.Talla (2023). Institutional Quality and Education Quality in Developing Countries. *Journal of the Knowledge Economy*, 14, 86–115. https://doi.org/10.1007/s13132-021-00869-9

Bydžovská, H. (2016). A Comparative Analysis of Techniques for Predicting Student Performance. *International Educational Data Mining Society*.

Costa, E.B., Fonseca, B., Santana, M.A., de Araújo, F.F., Rego, J. (2017). Evaluating the effectiveness of

educational data mining techniques for early prediction of students' academic failure in introductory programming courses. *Computers in Human Behavior*, 73, 247–256.

Erika B. Varga, A.S. (2021). Detecting at-risk students in Computer Science bachelor programs based on pre-enrollment characteristics. *Hungarian Educational Research Journal*, 11, 297–310. https://doi.org/10.1556/063.2021.00017

Fernandes, E., Holanda, M., Victorino, M., Borges, V., Carvalho, R., Van Erven, G. (2019). Educational data mining: Predictive analysis of academic performance of public school students in the capital of Brazil. *Journal of Business Research*, 94, 335–343.

Hashim, A.S., Awadh, W.A., Hamoud, A.K. (2020). Student Performance Prediction Model based on Supervised Machine Learning Algorithms. In: *IOP Conference Series: Materials Science and Engineering* (Vol. 928), p. 032019. IOP Publishing.

Helal, S., Li, J., Liu, L., Ebrahimie, E., Dawson, S., Murray, D.J., Long, Q. (2018). Predicting academic performance by considering student heterogeneity. *Knowledge-Based Systems*, 161, 134–146.

Hoffait, A.-S., Schyns, M. (2017). Early detection of university students with potential difficulties. *Decision Support Systems*, 101, 1–11.

Injadat, M., Moubayed, A., Nassif, A.B., Shami, A. (2020a). Multi-split optimized bagging ensemble model selection for multi-class educational data mining. *Applied Intelligence*, 50(12), 4506–4528.

Injadat, M., Moubayed, A., Nassif, A.B., Shami, A. (2020b). Systematic ensemble model selection approach for educational data mining. *Knowledge-Based Systems*, 200, 105992.

Iqbal, Z., Qadir, J., Mian, A.N., Kamiran, F. (2017). Machine learning based student grade prediction: A case study. *arXiv preprint arXiv:1708.08744*.

Jia, P., Maloney, T. (2015). Using predictive modelling to identify students at risk of poor university outcomes. *Higher Education*, 70(1), 127–149.

Karagiannopoulou, E., Milienos, F.S., Rentzios, C. (2021). Grouping learning approaches and emotional factors to predict students' academic progress. *International Journal of School & Educational Psychology*, 1–18.

Keser, S.B., Aghalarova, S. (2021). HELA: A novel hybrid ensemble learning algorithm for predicting academic performance of students. *Education and Information Technologies*, 1–32.

Kukkar, A., Mohana, R., Sharma, A., Nayyar, A. (2023). Prediction of student academic performance based on their emotional wellbeing and interaction on various e-learning platforms. *Education and Information Technologies*, 1–30.

Mai, T.T., Bezbradica, M., Crane, M. (2022). Learning behaviours data in programming education: Community analysis and outcome prediction with cleaned data. *Future Generation Computer Systems*, 127, 42–55. https://doi.org/10.1016/j.future.2021.08.026

Marbouti, F., Diefes-Dux, H.A., Madhavan, K. (2016). Models for early prediction of at-risk students in a course using standards-based grading. *Computers & Education*, 103, 1–15.

Martínez-Navarro, Á., Verdú, E., Moreno-Ger, P. (2021). *Mining Pre-Grade Academic and Demographic Data to Predict University Dropout*. Springer, Singapore, pp. 197–215. https://doi.org/10.1007/978-981-16-3941-8_11

Miguéis, V.L., Freitas, A., Garcia, P.J., Silva, A. (2018). Early segmentation of students according to their academic performance: A predictive modelling approach. *Decision Support Systems*, 115, 36–51.

Nti, I.K., Akyeramfo-Sam, S., Bediako-Kyeremeh, B., Agyemang, S. (2022). Prediction of social media effects on students' academic performance using Machine Learning Algorithms (MLAs). *Journal of Computers in Education*, 9(2), 195–223.

Qazdar, A., Er-Raha, B., Cherkaoui, C., Mammass, D. (2019). A machine learning algorithm framework for predicting students performance: A case study of baccalaureate students in Morocco. *Education and Information Technologies*, 24(6), 3577–3589.

Sandoval, A., Gonzalez, C., Alarcon, R., Pichara, K., Montenegro, M. (2018). Centralized student performance prediction in large courses based on low-cost variables in an institutional context. *The Internet and Higher Education*, 37, 76–89.

Tan, C.J., Lim, T.Y., Liew, T.K., Lim, C.P. (2022). An intelligent tool for early drop-out prediction of distance learning students. *Soft Computing*, 1–17.

Thiele, T., Singleton, A., Pope, D., Stanistreet, D. (2016). Predicting students' academic performance based on school and socio-demographic characteristics. *Studies in Higher Education*, 41(8), 1424–1446.

Tight, M. (2020). Student retention and engagement in higher education. *Journal of Further and Higher Education*, 44(5), 689–704. https://doi.org/10.1080/0309877X.2019.1576860

Tomasevic, N., Gvozdenovic, N., Vranes, S. (2020). An overview and comparison of supervised data mining techniques for student exam performance prediction. *Computers & Education*, 143, 103676.

Veloso, B., Barbosa, M.A., Faria, H., Marcondes, F.S., Durães, D., Novais, P. (2023). A Systematic Review on Student Failure Prediction. In: *International Conference in Methodologies and intelligent Systems for Techhnology Enhanced Learning*, pp. 43–52. Springer.

Wang, X., Zhao, Y., Li, C., Ren, P. (2023). ProbSAP: A comprehensive and high-performance system for student academic performance prediction. *Pattern Recognition*, 137, 109309. https://doi.org/10.1016/j.patcog.2023.109309

Wild, S., Rahn, S., Meyer, T. (2023). The relevance of basic psychological needs and subject interest as explanatory variables for student dropout in higher education—a German case study using the example of a cooperative education program. *European Journal of Psychology of Education*, 1–18.

Xing, W., Guo, R., Petakovic, E., Goggins, S. (2015). Participation-based student final performance prediction model through interpretable Genetic Programming: Integrating learning analytics, educational data mining and theory. *Computers in Human Behavior*, 47, 168–181.

Xu, J., Moon, K.H., Van Der Schaar, M. (2017). A machine learning approach for tracking and predicting student performance in degree programs. *IEEE Journal of Selected Topics in Signal Processing*, 11(5), 742–753.

Xu, X., Wang, J., Peng, H., Wu, R. (2019). Prediction of academic performance associated with internet usage behaviors using machine learning algorithms. *Computers in Human Behavior*, 98, 166–173.

Yağcı, M. (2022). Educational data mining: prediction of students' academic performance using machine learning algorithms. *Smart Learning Environments*, 9(1), 11.

Yao, H., Lian, D., Cao, Y., Wu, Y., Zhou, T. (2019). Predicting academic performance for college students: A campus behavior perspective. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(3), 1–21.

Yousafzai, B.K., Hayat, M., Afzal, S. (2020). Application of machine learning and data mining in predicting the performance of intermediate and secondary education level student. *Education and Information Technologies*, 25(6), 4677–4697.

Z. Alamgir is currently an Associate professor at the Department of Computer Science, NUCES Lahore. In her PhD., she has developed combinatorial generation algorithms for subgraphs that can efficiently detect isolated communities in the real world. Her research interests include Big Data, Recommendation Systems, Data Mining, Distributed Computing, and Combinatorial Algorithms. She has established a Big Data Lab at NUCES. Currently, many postgraduate and graduate students are conducting research in the Bigdata lab using emerging cluster computing frameworks like Apache Spark and Federated Learning.

H. Akram has done her MS in Computer science and is working as Cyber Security Analyst.

S. Karim obtained her PhD in Algorithms. Her research interests include Social Network Analysis, Data Science and Distributed Computing.

A. Wali has taught at the National University of Computer and Emerging Sciences since 2004. His areas of interest include Machine Learning, Image Processing, Natural language processing, Virtual/Augmented Reality, and Font Development.