Predicting Academic Performance in Computer Science Department Using Machine Learning Techniques and Analysis of Students' Academic Statements

Srinivas Dandu
Narsimha Reddy Engineering College

Predicting Academic Performance in Computer Science Department Using Machine Learning Techniques and Analysis of Students' Academic Statements

Dandu Srinivas[1], Nakka Venkatesh[2] and M Prabhakar[3]

[1] Assistant Professor, Department of CSE, NRCM
[2,3] Assistant Professor, Department of CSE, CMREC

Abstract. In the realm of higher education, understanding the factors that contribute to academic achievement remains a significant area of research. This paper aims to provide insights into forecasting the educational performance of the Computer Science Department through the employment of machine learning techniques and analysis of students' academic statements. The growing demand for accurate predictions regarding academic achievement has prompted researchers to explore novel approaches that leverage the vast amount of available data. By utilizing machine learning algorithms and examining students' academic records, valuable patterns and trends can be identified, enabling accurate predictions and facilitating proactive measures to enhance student performance. This paper discusses the methodology, dataset, and results of our study, highlighting the potential benefits and implications of utilizing machine learning in the domain of educational forecasting.

Keywords: Education, Prediction, Student Performance, Machine Learning Models, Educational Analytics.

1 Introduction

The growing availability of digital data in the education sector presents an opportunity to gain deeper insights into student performance. This paper focuses on the application of machine learning algorithms to extract meaningful patterns and predictive models from students' academic statements within the Computer Science Department. By harnessing the power of data analytics, institutions can identify key factors that influence academic achievement and design targeted interventions accordingly.

The Computer Science Department is witnessing a surge in student enrollment, creating a need for efficient methods to anticipate academic performance. In this section, we provide an overview of the significance of forecasting educational outcomes and the potential benefits of employing machine learning techniques for this purpose. This research focuses on identifying areas for improvement within a computer science department and its students. By analyzing academic statements, the aim is to predict student performance in different fields. Academic statements refer to the students' academic results, and the courses offered by the computer science department are categorized into fifteen fields. These fields represent potential future professional areas where students may work.

Benefits and Significance: Solving this problem offers multiple benefits. Firstly, it assists the department in preparing students for the competitive job market and higher studies. By pinpointing areas of weakness, the department can provide additional support and resources to strengthen students' skills in those fields. Secondly, it enhances the department's reputation by ensuring that graduates are well-equipped for their professional careers.

Implementation: Multiple machine learning algorithms, specifically the multiclass-multioutput classification algorithm, are utilized to implement the proposed model. The academic areas of the department are divided into fifteen fields, each categorized as Excellent, Very Good, Good, Average, or Bad. A dataset comprising one thousand students' academic statements is used to develop and compare different models. Algorithms such as Decision Tree, Extra Tree, Extra Trees, K-Neighbors, Radius Neighbors, and Random Forest Classifier are employed. Evaluation metrics, including Accuracy, Precision, Recall, and F1 score, are used to assess the effectiveness of the trained models. The Random Forest Classifier exhibits the best performance among all implemented ML algorithms, based on these evaluation metrics.

2 Related Work

Machine Learning techniques to predict academic outcomes based on various factors, including demographic information, academic records, and student behaviors. However, limited attention has been given to utilizing model-based machine learning to forecast the performance of a computer science department.

Evaluation of Learning Outcomes using Academic Data Analysis: El-Halees [8] conducted a study that involved analyzing academic data to evaluate learning outcomes.

Forecasting Academic Performance through Feature Space Analysis: Ghassen Ben Brahim [9] focused on classifying students as strong or poor performers by generating an 86-dimensional feature space using informative features.

Evaluation of Exam Performance using Machine Learning Techniques: Nikola et al. [10] employed various machine learning techniques to assess the effectiveness of their proposed strategy in predicting exam performance.

Machine Learning Approaches for Student Performance Analysis: Sekeroglu et al. [11] explored the use of diverse machine learning approaches for analyzing the Student Performance Dataset and the Student's Academic Performance Dataset.

Decision Trees for Predicting Learning Outcomes and Enrollment Management: Yadav et al. [12] utilized decision trees to forecast learning outcomes and manage student enrollment. Decision trees provided easily interpretable classification rules and accurate predictions, aiding in identifying students who needed special attention and potentially reducing the dropout rate.

Model for Classifying Students Based on Academic Achievement: Mesarić et al. [13] developed a model that incorporated high school achievement data, program outcomes after the first year of study, and teacher recommendations to classify students based on academic achievement at the end of their first academic year.

Prediction of Academic Performance using Random Forest and SMOTE: Rastrollo-Guerrero et al. [14] employed a combination of random forest and SMOTE oversampling to analyze students' academic performance and engagement with a virtual learning environment. The study revealed that the random forest classifier effectively predicted academic performance, particularly when more accurate data became available toward the end of the course presentation.

3 Methodology

3.1 Data Collection and Preprocessing

The initial step involves gathering the data, which is then transformed into comma-separated values for further analysis. To ensure accuracy and completeness, the actual results and grading system for each course in the dataset are provided in the Appendix section. Any partial, inaccurate, corrupted, or poorly formatted data is processed to ensure consistency.

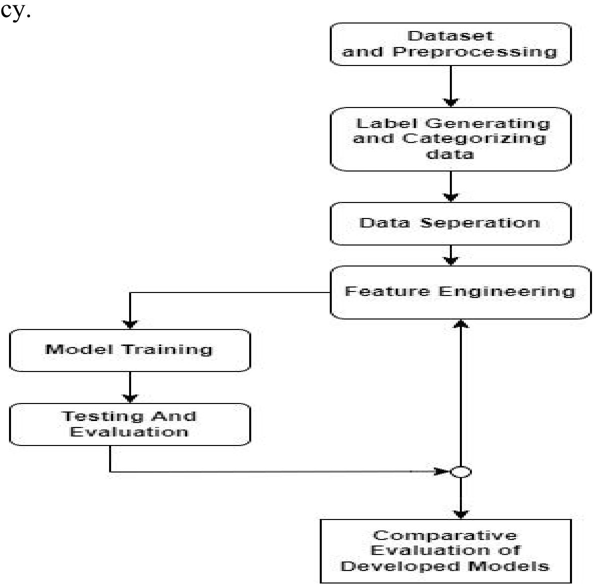
Dataset and Preprocessing
Label Generating and Categorizing data
Data Seperation
Feature Engineering
Model Training
Testing And Evaluation
Comparative Evaluation of Developed Models

Fig. 1 Working Diagram for Machine Learning Model

3.2 Label Generation and Categorization

To generate labels or class columns from the students' academic results, the mean value of each category among the fifteen fields is calculated and assigned as the output column. The grade point range determines the grouping of values into specific performance categories, including excellent, very good, good, weak, and bad. These categories serve as indicators of the students' academic performance.

Experimental Environment and Data Preparation: For conducting the prediction analysis, this research utilizes Google Collaboratory, a web-based Python IDE, as the experimental environment. Table 2 represents the grade point range and the corresponding performance categories. To enhance the performance of the model, a Label Encoder is applied to normalize each label.

Multiclass and Multioutput Problem: The nature of the problem at hand is a multiclass and multioutput problem since it requires predicting values for multiple columns, each containing multiple classes. In this stage, the data is divided into training and testing sets, with 70% of the data allocated for training purposes and the remaining 30% for testing the model's performance. This partitioning ensures a robust evaluation of the predictive model.

Enhancing Model Accuracy through Feature Engineering: To improve the accuracy of the model, feature engineering techniques were employed. This process involves transforming the raw, unprocessed data into meaningful features that can be effectively utilized in the machine learning methods employed in this research. The dataset undergoes the four essential phases of feature engineering, namely feature creation, modification, extraction, and classification. These phases contribute to enhancing the overall predictive power of the model.

Model Configuration and Selection of Machine Learning Techniques: A critical aspect of developing a successful machine learning model is configuring the model appropriately to address the specific problem at hand. In this research, we carefully select a suitable machine learning technique that aligns with the objective of accurately forecasting our intended goals. Given the nature of the problem as a multiclass-multioutput classification, we applied various relevant machine learning algorithms, including the Decision Tree Classifier (tree), Extra Tree Classifier (ensemble), Extra Trees Classifier (neighbors), K Neighbors Classifier (neighbors), Radius Neighbors Classifier (ensemble), and Random Forest Classifier (ensemble).

Training, Testing, and Prediction: With the training and testing data properly separated, we proceed to perform the crucial steps of training, testing, and prediction. The model is trained on the training data, enabling it to learn the underlying patterns and relationships within the dataset. Subsequently, the model is tested on the testing data to assess its performance and evaluate its ability to make accurate predictions. Finally, utilizing the trained model, predictions are generated for the desired outcomes.

3.3 Comparative Evaluation of Developed Models

To comprehensively assess the performance of each developed model, a comparison of all the evaluation metrics is conducted. These evaluation metrics provide valuable insights into the effectiveness and efficiency of the different models. By considering metrics such as accuracy, precision, recall, and other relevant evaluation criteria, we gain a comprehensive understanding of the strengths and weaknesses of each model, facilitating the identification of the best-performing approach.

4 Dataset

In this data collection process, including the types of academic statements considered and the anonymization procedures implemented to ensure privacy. Additionally, we discuss the characteristics of the dataset and any potential biases or limitations associated with it.

The dataset used in this research contains academic records of 1,000 students from the Computer Science and Engineering Department. The grading system used in the dataset ranges from 0.00 to 4.00, and letter grades are assigned based on a student’s numerical grade. A+ is assigned to numerical grades of 80% and above, while F is assigned to numerical grades of less than 40%. The remaining letter grades (A, A-, B+, B, B-, C+, C, and D) are assigned to numerical grades that fall between these two extremes. The grading system which is used for our dataset of each course is presented in the Appendix section. Each row in the dataset represents a student's academic performance throughout their study in the department. Simply put, in our dataset, the columns represent the courses that were considered for our research, and each row represents the result of those courses for each student. After processing and refining the incomplete, inaccurate, corrupted, or poorly formatted data, a total of 1,000 student records are included in our final dataset. This dataset was used to develop a machine learning model to predict the performance of the department's students and identify areas that need improvement.

5 Machine Learning Algorithms

We present various machine learning algorithms utilized in our study for predicting educational performance. This includes regression models, classification models, and ensemble techniques. The rationale behind the selection of these models and their implementation details are discussed, along with the evaluation metrics employed to assess their performance.

A. Decision Tree Classifier

The Decision Tree Classifier is a widely-used supervised learning algorithm employed for classification tasks. It operates by partitioning the dataset into subsets based on the most influential attributes, employing a decision tree structure to depict decisions and their potential outcomes. Through an iterative process, the algorithm recursively divides the data into smaller subsets using the attribute that yields the most optimal split.

Information Gain: Information Gain serves as a measure to quantify the level of homogeneity in the dataset before and after a split. This metric is calculated using the following formula:

Information Gain = Entropy(parent) - [Weighted Average] Entropy(children)

Entropy, another crucial concept, measures the randomness or impurity within a dataset. The formula for entropy is as follows:

Entropy(S) = -p1log2(p1) - p2log2(p2) - ... - pn*log2(pn)

Here, p1, p2, ..., pn represent the proportions of examples belonging to each class within the dataset.

Gini Impurity:Gini Impurity is an alternative measure of impurity or randomness in a dataset. It is calculated using the following formula:

Gini(S) = 1 - p1^2 - p2^2 - ... - pn^2

Similar to entropy, p1, p2, ..., pn represent the proportions of examples belonging to each class within the dataset.

Pruning:Pruning is a valuable technique employed to mitigate overfitting in decision tree algorithms. It involves selectively removing branches from the tree that do not enhance the model's accuracy on the test data. By eliminating unnecessary branches, pruning helps ensure a more robust and generalized decision tree model.

B. K-Nearest Neighbors (K-NN) Classifier

The K-Nearest Neighbors (K-NN) algorithm is a non-parametric and lazy learning algorithm employed for classification and regression tasks. It classifies data points based on the classes of their K nearest neighbors. The value of K is determined by the user. The K-NN algorithm is straightforward to understand and implement. It makes no assumptions about the data distribution and can handle both binary and multi-class classification tasks. However, the algorithm can be computationally expensive and may face challenges with high-dimensional data, known as the curse of dimensionality. In classification, the K-NN algorithm operates as follows:

I. Calculate the distance between the test instance and all training instances.

II. Select the K nearest instances based on distance.

III. Determine the class of the test instance based on the majority class of its K nearest neighbors.

C. Radius Neighbors (Neighbors) Classifier

The Radius Neighbors Classifier is a machine learning algorithm used for classification tasks. Unlike the K-Nearest Neighbors algorithm, it classifies observations based on the class of their neighbors within a fixed radius, rather than a fixed number of nearest neighbors. The radius is defined by the user, and any observation outside the radius is disregarded. The classification decision is made based on the majority class of the neighbors within the radius. In case of a tie, the class of the closest neighbor is chosen as the classification. The equation for the Radius Neighbors Classifier is similar to that of the K-Neighbors Classifier, but it considers all neighbors within a fixed radius "r" instead of a specific number of neighbors.

D. Random Forest (Ensemble) Classifier

Random Forest is an ensemble learning method that utilizes multiple decision trees to make predictions. Each decision tree in the forest is constructed using a random subset of the training data and a random subset of features. This randomness helps alleviate overfitting and enhances the model's generalization capability. During the prediction phase, each tree independently makes a prediction, and the final prediction is based on the majority vote of all the trees. In the training phase, multiple decision trees are built using different subsets of the data and features. In the prediction phase, the output is determined by the majority vote of all the decision trees [25]. In this research, since the academic areas are divided into multiple categories, a multiclass-multioutput classification algorithm like the random forest classifier is employed to make predictions. The results demonstrate that the random forest classifier achieves the highest accuracy.

6 Experimental Results

The experimental results and performance evaluation of the machine learning models applied to predict academic performance. The accuracy, precision, recall, and other relevant metrics are reported, along with comparisons between different models and approaches. Furthermore, we discuss the insights derived from the analysis of the results.

The courses are divided into fifteen categories. Each category’s performance is broken down into five sections. Excellent, Very Good, Good, Average, and Bad are the titles of the sections. Figure 2., below compares model accuracy scores in terms of percentage. The model developed by the DTC algorithm accurately predicts the labels 91% of the time. The model created by ETC and ETCE has an accuracy rating of almost 92%, RNC is nearly 70%, RFCE is around 94%, and KNC is almost 84%. The RFCE has the highest accuracy score among the many models using multiclass-multioutput machine learning algorithms. Therefore, this model more accurately predicts the performance of the fifteen categories of these departments.

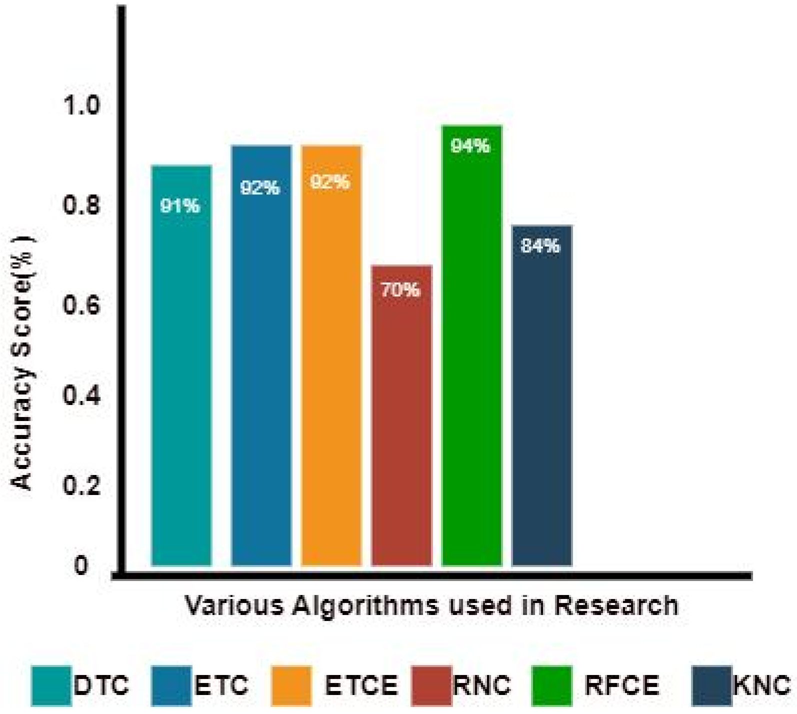

Figure 2. Model accuracy score in percentage (%)

7 Conclusion

In conclusion, this paper demonstrates the value of employing machine learning techniques and analyzing students' academic statements for predicting the educational performance of the Computer Science Department. The insights gained from this research can be instrumental in implementing proactive measures to improve student outcomes and facilitate personalized educational interventions.

References

1. Sudais M, Safwan M, Khalid MA, Ahmed S. Students’ academic performance prediction model using machine learning -2022.
2. Yağcı M. Educational data mining: Prediction of students’ academic performance using machine learning algorithms. Smart Learning Environments. 2022;
3. Dervenis C, Kyriatzis V, Stoufis S, Fitsilis P. Predicting students’ performance using machine learning algorithms. Proceedings of the 6th International Conference on Algorithms, Computing and Systems.
4. Khan MAR, Akter J, Ahammad I, Ejaz S, Jaman Khan T. Dengue outbreaks prediction in Bangladesh perspective using distinct multilayer perceptron NN and decision tree. Health Information Science & System. 2022;
5. Khan MAR, Afrin F, Prity FS, Ahammad I, Fatema S, Prosad R, et al. An effective approach for early liver disease prediction and sensitivity analysis. Iran Journal of Computer Science. 2023.
6. Khan AR, Uzzaman F, Ahammad I, Prosad R, Zayed-Us-salehin, Khan TZ, et al. Stock market prediction in Bangladesh perspective using artificial neural network. International Journal of Advanced Technology and Engineering Exploration. 2022.
7. Poudyal S, Mohammadi-Aragh MJ, Ball JE. Prediction of student academic performance using a hybrid 2D CNN model. Electronics. 2022.
8. El-Halees AM. Mining students’ data to analyze e-Learning behavior: A case study. Computer Science, Education; 2009.

9. Brahim GB. Predicting student performance from online engagement activities using novel statistical features. Arabian Journal for Science and Engineering. 2022.
10. Tomasevic N, Gvozdenovic N, Vranes S. An overview and comparison of supervised data mining techniques for student exam performance prediction. Computers & Education. 2020.
11. Sekeroglu B, Dimililer K, Tuncal K. Student performance prediction and classification using machine learning algorithms. Proceedings of the 2019 8th International Conference on Educational and Information Technology, 2019.
12. Yadav SK, Bharadwaj B, Pal S. Mining education data to predict student's retention: A comparative study. International Journal of Computer Science and Information Security. 2012.
13. Mesarić J, Šebalj D. Decision trees for predicting the academic success of students. Croatian Operational Research Review. 2016.
14. Rastrollo-Guerrero JL, Gómez-Pulido JA, Durán-Domínguez A. Analyzing and predicting students' performance by means of machine learning: A review, 2020.
15. Shaik AB, Srinivasan S. A brief survey on random forest ensembles in classification model. In: Bhattacharyya S, Hassanien A, Gupta D, Khanna A, Pan I. (eds.) International Conference on Innovative Computing and Communications. Lecture Notes in Networks and Systems, vol 56. Springer, Singapore; 2018.