International Journal of Computing Sciences Research (ISSN print: 2546-0552; ISSN online: 2546-115X)
Vol. 9, pp. 3804-3819
doi: 10.25147/ijcsr.2017.001.1.246
https://stepacademic.net

Short Paper

Predicting At-Risk Students' Academic Performance in an Online Learning Environment Using Learning Management System Interaction Data and the Random Forest Algorithm

Ali A. Naim
University of Makati, Philippines
ali.naim@umak.edu.ph

Date received: July 30, 2024
Date received in revised form: August 3, 2024; January 27, 2025
Date accepted: May 8, 2025

Recommended citation:

Naim, A. A. (2025). Predicting at-risk students' performance in an online learning environment using learning management system interaction data and the random forest algorithm. *International Journal of Computing Sciences Research*, 9, 3804-3819. https://doi.org/ 10.25147/ijcsr.2017.001.1.246

Abstract

Purpose – This study aims to develop a model to identify at-risk students from LMS interaction data, analyzing how existing machine learning models can improve this identification.

Method – A machine learning model was created using five classifiers: random forest (RF), support vector machine, Naive Bayes, logistic regression, and K-nearest neighbor, to predict student performance from LMS interactions in a dataset of 486 students from a local university.

Results – The Random Forest algorithm achieved an MCC score of 66.42%, a Kappa score of 64.94%, and an F1 score of 66.62%.

Conclusion – LMS has enhanced education by improving accessibility and centralizing information, but challenges remain in identifying at-risk students. ML models like Random Forest show promise in addressing this issue.

Recommendations – Use more reliable datasets, explore imbalance treatment techniques, and integrate Random Forest predictive modeling to identify at-risk students in LMS.
Research Implications – This research seeks to promote the use of robust methods for improving predictive modeling accuracy using Random Forest to identify at-risk students.

Practical Implications – This research provides insights on predictive modeling using Random Forest and student interaction data from LMS to enable timely interventions and improve student success and learning outcomes.

Keywords – student performance, online learning, education, Learning Management Systems (LMS), predictive modeling, random forest, machine learning (ML)

INTRODUCTION

The 21st century is marked by technological advancements, greatly impacting education. Technology has transformed education, especially during the COVID-19 pandemic, leading to a rise in online learning (Raja & Nagasubramani, 2018; Ulfa & Fatawi, 2021). Higher education institutions must ensure quality learning experiences, as student attrition remains a significant concern influenced by various factors. Academic success is crucial in addressing attrition, and institutions provide interventions to improve it (Beer & Lawson, 2016). Identifying at-risk students is the first step, though it is often manual and biased (Imran et al., 2019). Early identification allows educators to provide timely support (Zhang et al., 2014). Predicting student performance offers insights into dropouts and learning outcomes, helping institutions adjust teaching methods and adopt new LMS (Brahim, 2022). Technologies like LMS and machine learning can aid in this process.

A learning management system (LMS) is an online platform used to administer educational programs, mimicking traditional classrooms. It includes tools for synchronous and asynchronous communication, management features, and evaluation utilities (Lopes, 2014). These features facilitate course structure and enable self-directed learning for professors and students (Toro & Reischl, 2018). LMS logs capture data on student interactions, such as access times, durations, and frequencies for content, quizzes, forums, and other activities. This unique data for each learner (Imran et al., 2019) can be used to develop predictive models to enhance learning processes like evaluation and counseling. Various machine learning (ML) algorithms have been applied to student demographic, socioeconomic, pre-enrollment, enrollment, academic, and LMS data to predict academic progress (Shahiri et al., 2015; Conijn et al., 2017). This study aims to develop a predictive model to identify at-risk students from an LMS interaction dataset, analyzing how existing ML approaches can achieve this goal.

LITERATURE REVIEW

Foreign Literature

Research globally highlights the importance of Learning Management Systems (LMS) in enhancing educational experiences, especially during crises like the COVID-19 pandemic. LMS is essential for continuing education when traditional classrooms are unavailable (Wibawa et al., 2021; Shirinkina, 2022). Integrating Learning Analytics (LA) and Educational Data Mining (EDM) allows for the extraction of valuable insights from large datasets, improving learning outcomes and personalizing education (Santos et al., 2023).

Foreign Studies

Studies show that predictive analytics through LMS can identify at-risk students early, enabling targeted interventions to support academic progress (Aguirre & Legaspi, 2020; Warnars et al., 2020). Factors like student attitudes, reflective thinking, problem-solving skills, and teacher-related factors significantly influence academic performance (Macaso & Dagohoy, 2022). These insights help educational institutions tailor teaching methods and support systems to meet diverse student needs.

Local Literature

Research in the Philippines emphasizes the importance of predicting student performance to improve educational outcomes. Factors like attitude towards Mathematics, reflective thinking, problem-solving skills, and parental involvement significantly influence academic performance (Buctot et al., 2021). Predictive analytics through LMS can identify at-risk students early, enabling targeted interventions (Balasico & Tan, 2020). Using LMS interaction data and the Random Forest algorithm can greatly enhance student performance predictions. Leveraging tree-based ensemble methods like Random Forest, XGBoost, and LightGBM allows educational institutions to optimize predictions and identify at-risk students early (Ayulani et al., 2023).

Local Studies

Implementing machine learning and deep learning models has shown high accuracy in predicting student outcomes, aiding informed decision-making, and improving educational results (Martínez-Martínez et al., 2023). Machine learning techniques are increasingly used to predict student performance and identify factors influencing low achievement in science education (Bernardo et al., 2023). Algorithms such as logistic regression, random forest, and support vector machines analyze student performance data, offering valuable insights for educators and parents (Kaur et al., 2023). These methods help identify at-risk students early, enabling timely interventions and support.

This study explores the growing field of predicting student performance in e-learning environments. The proponent used five machine learning algorithms: random forest (RF), support vector machine, logistic regression, K-nearest neighbor, and Naive Bayes. The dataset was extracted from an LMS of a local university. This repository includes datasets used in the laboratory's research on predicting student performance and engagement in e-learning settings.

Conceptual Framework

The development of the Random Forest model involves several stages. First, the dataset is collected from an LMS. During preprocessing, steps like standardization, SMOTE, and encoding of categorical data are performed (Figure 1). The data is split into training and testing sets. The training data fits the Random Forest and other machine learning models. Finally, the testing data evaluates the models' performance using MCC, F1 Score, and Cohen’s Kappa.

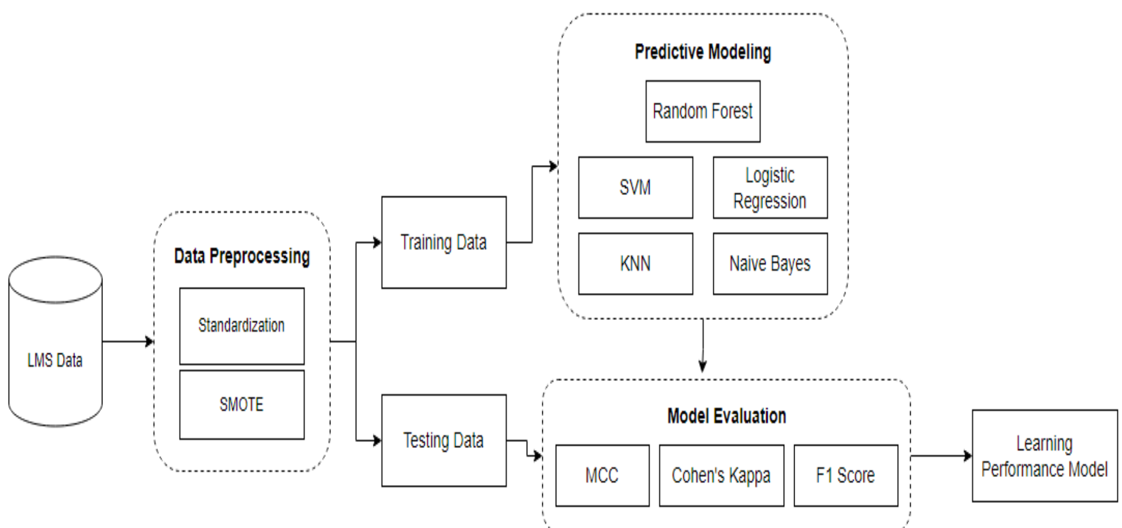

Figure 1. Conceptual Framework of the Study

METHODOLOGY

This chapter details the study’s methodology, including data gathering, preprocessing, model creation, and evaluation. Statistical techniques and predictive modeling were used to analyze and process the data. The following subsections describe the algorithms and technologies employed.

Data Gathering and Pre-processing

Data was collected from an LMS of a local university, detailing interactions of 486 students with an LMS during an undergraduate science course. Student performance is classified as Good (60-100%) or Weak (<59%). The dataset includes LMS interaction data, such as assignment times, scores, and login frequency. It is split into 70% training and 30% testing sets. Although the Random Forest model needs minimal preprocessing (Breiman, 2001), steps like encoding ordinal values, creating synthetic data, and scaling are essential for quality results. These steps will be detailed in the following subsections.

Scaling

Robust Scaler standardizes dataset features to reduce value ranges, performing better than other scalers when outliers are present (Pedregosa et al., 2011). It scales data into a normal distribution by removing the median and Interquartile Range (IQR), which is the difference between the 75th and 25th percentiles. Given the presence of outliers in the dataset, Robust Scaler is the preferred scaling technique. The formula for Robust Scaler is as follows:

$$X_{scale} = \frac{X_i - X_{med}}{IQR} \qquad \textit{Equation 1}$$

Where Xi is the value of each feature, Xmed is the median, and IQR is the Interquartile Range.

Oversampling

Synthetic Minority Over-Sampling Technique (SMOTE) addresses imbalanced datasets by generating synthetic data to balance class distributions (Chawla et al., 2002). Using Accuracy to measure predictive models on imbalanced datasets can lead to biases, so metrics like Precision and Recall are often more appropriate. In this study, SMOTE is used to balance classes, allowing Accuracy to be a valid metric. SMOTE focuses on the minority class, generating synthetic data based on its nearest neighbors, which improves results, especially with tree-based models like Random Forest (Quinlan, 1987). Figure 2 illustrates the SMOTE algorithm for imbalanced datasets:

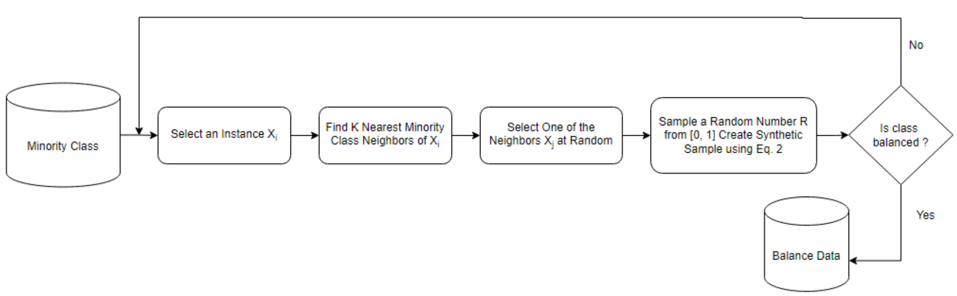

Figure 2. Flowchart of SMOTE

Figure 2 shows the flowchart of the SMOTE algorithm, which focuses on the minority class and performs a process to generate synthetic data. First, it picks a random instance of the minority class. Then, it finds the nearest neighbour of that data. After that, it performs the calculation of synthetic data using equation 2. Lastly, it repeats the first step until the data is balanced. The formula for creating the synthetic data is denoted by:

$$X_k = X_i + (R * X_i - X_j) \qquad \textit{Equation 2}$$

where Xk is the generated synthetic sample, R is a random number from either 0 or 1, Xi is the original instance, and Xj is a nearest neighbor of Xi.

Predictive Modelling Using Random Forest

Decision Trees

Random Forest is an ensemble model of several Decision Trees, known as weak learners due to their limited nodes and low computational power (Breiman, 2001). The model's performance and complexity depend on the number of decision trees. Decision trees are created using Attribute Selection Methods. In this study, with primarily continuous features, the Gini Index is used to identify splits for each tree. The attribute with the smallest Gini Index value becomes the root node (Sundhari, 2011). The Gini Index is denoted by Equation (3).

$$Gini(t) = 1 - \sum_{i=1}^{j} P(i|t)^2 \qquad \textit{Equation 3}$$

where j represents the number of classes in the label, while P represents the ratio of the class at the ith node.

Bagging

Random Forest uses bagging, with each weak learner processing input data independently. The data undergoes bootstrapping, partitioning into subsets. Two-thirds train the model, while one-third serves as validation (Out Of Bag Score) (Breiman, 2001). Each weak learner predicts a nominal variable, and the final prediction is determined by voting, with the most votes as the final output. A robust score depends on the number of weak learners, which increases computational power.

Model Evaluation

Model Evaluation is the process of quantifying the model’s performance based on different metrics. To accomplish this, the developed model will be evaluated on the test data. This is to ensure that the model is performing well. The primary evaluation metrics for this study are further discussed in the following subsections.

Matthews correlation coefficient

Matthews' correlation coefficient (MCC) is a metric for evaluating classification problems. It takes into account all the variables in the confusion matrix, and it can also be used for imbalanced datasets (Matthews, 1975; Gorodkin, 2004). The formula is denoted by:

$$MCC = \frac{TPxTN - FPxFN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \qquad \textit{Equation 4}$$

Where TP is the True Positive, Tn is the True Negatives, FP is the False Positives, and FN is the False Negatives.

Cohen's Kappa

Cohen's Kappa is a measure for quantifying the classification of a predictive model's performance based on the inter-rater reliability (Cohen, 1960; Artstein & Poesio, 2008). It measures the agreement of two raters who classify items. It is denoted by the equation:

$$K = \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e} \qquad \textit{Equation 5}$$

where Pe is the coincidence ratio resulting from randomness, and Po is the observed coincidence ratio.

F1-Score

F1 Score is the harmonic mean of precision and recall (Sasaki, 2007). It combines information presented by the precision and recall, which are vital for an imbalanced dataset. The formula for F1-Score is as follows:

$$F_1 = 2\frac{P\ x\ R}{P + R} \qquad \textit{Equation 6}$$

Where P is the Precision, and R is the Recall.

RESULTS

Training, Testing, and OOB Score

The evaluation of the five machine learning models is shown in Figures 3, 4, and 5, based on training and testing data. This process checks for overfitting or underfitting. Since accuracy is not suitable for imbalanced data, MCC, F1-Score, and Cohen's Kappa are used as primary metrics.

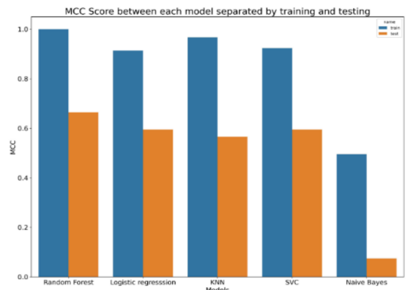

Figure 3. Train and Test Accuracy of ML models

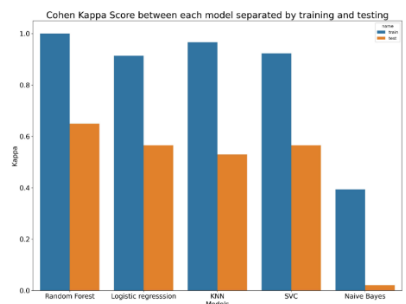

Figure 4. Train and Test Kappa of ML models

Figure 5. Train and Test F1 score of ML models

The evaluation results indicate that all models are overfitted, performing well on training data but poorly on test data. Hyperparameter tuning did not significantly alter the results compared to the baseline model. The distribution of classification metrics is similar, with Random Forest achieving the highest train score (100%), a test score (66%), and an

OOB score (98%). Logistic Regression and Support Vector Classifier have similar train and test scores and could benefit from further tuning. The Naive Bayes model underperformed, with train scores of 49% (MCC), 39% (Kappa), and 76% (F1 Score), and low test scores of 7%, 2%, and 9%, respectively, indicating underfitting.

Cross Validation

In this section, the model is evaluated using cross-validated data from the training set, divided into three folds. Figures 6, 7, and 8 show the model's performance across different folds using MCC, F1-Score, and Cohen's Kappa.

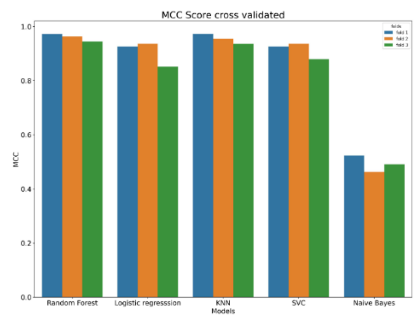

Figure 6. Cross-Validated Test Accuracy of ML models

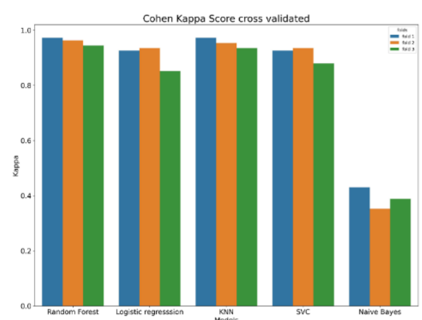

Figure 7. Cross-Validated Test Kappa of ML models

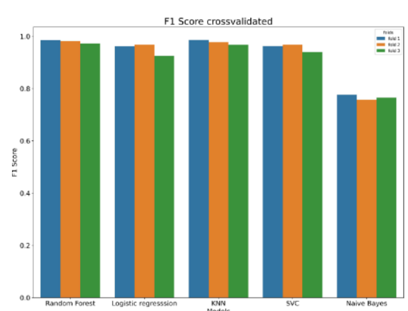

Figure 8. Cross-Validated Test F1 score of ML models

The cross-validation results indicate that the Random Forest model achieved the highest scores across all primary metrics, with a mean score of 96% in all folds. KNN was the second highest, with a mean score of 95%. Naive Bayes performed poorly, with a mean score of 49%, indicating it is unsuitable for the given problem and data.

Feature Importances and Confusion Matrix

This section investigates the model's behavior using the Random Forest model's built-in Feature Importances function, along with the confusion matrix, which includes True Positives, True Negatives, False Positives, and False Negatives.

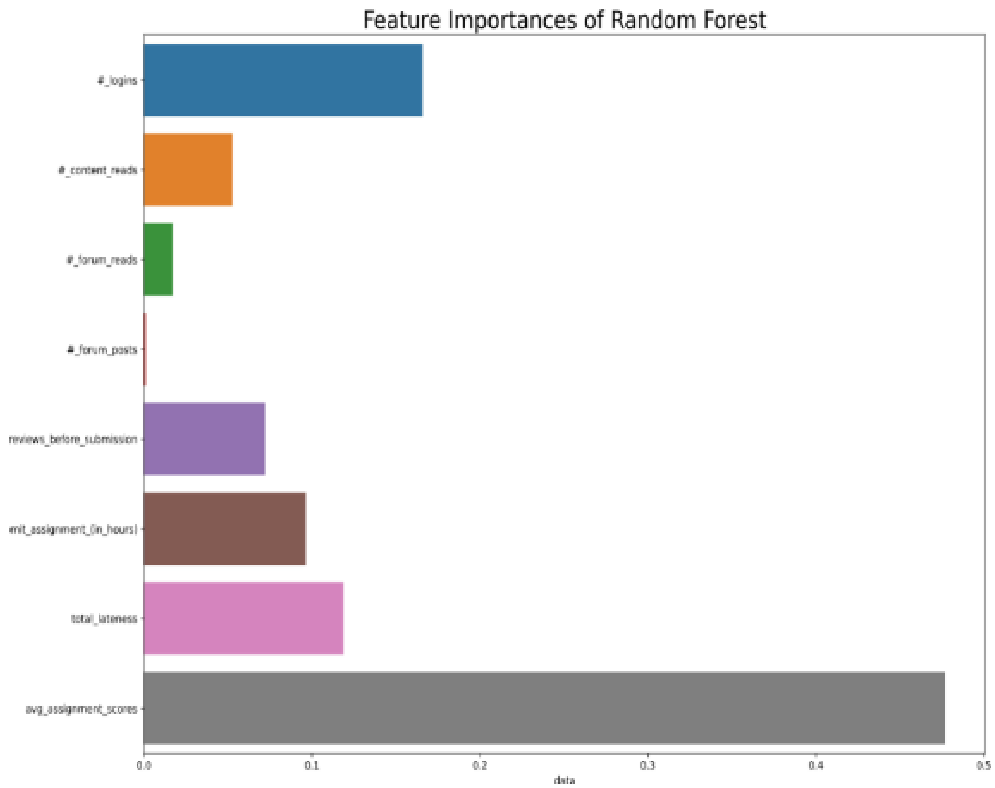

Figure 9. Feature Importances of the Random Forest Model

Figure 9 displays the Feature Importances of the Random Forest model. The y-axis shows the features (predictors), and the x-axis shows their importance. A higher importance score indicates a greater impact on predictions. The most important feature is the average assignment scores (*avg_assignment_scores*), followed by the number of logins (*#_logins*), which reflects how often the student has logged into the LMS.

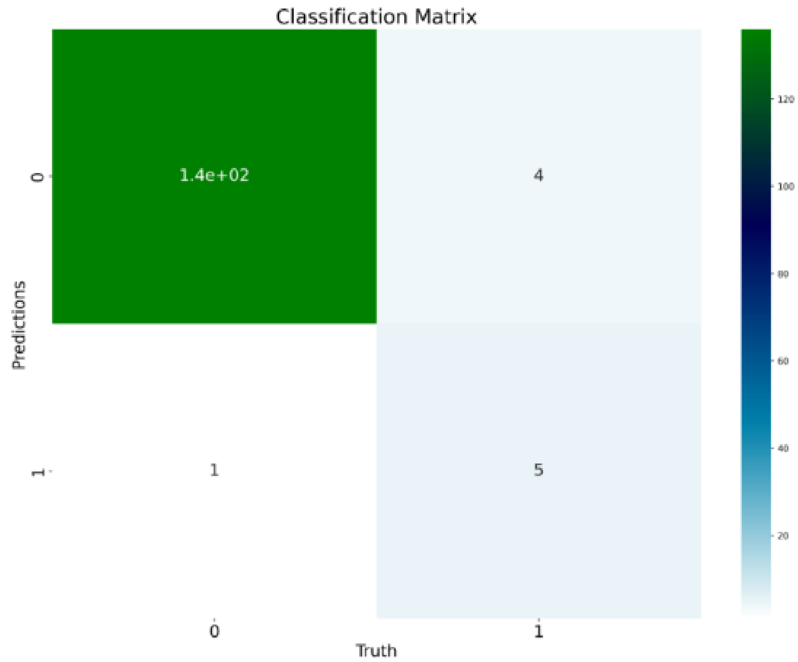

Figure 10. Confusion Matrix of the Random Forest

Figure 10 illustrates the confusion matrix for the Random Forest model. The x-axis represents the actual class (Truth), and the y-axis represents the model's prediction. Here, 1 denotes a weak learner, and 0 denotes a good learner. The matrix shows a notable class imbalance, with many True Positives and few True Negatives. There are also significant False Positives, which reduce the model's precision. While adjusting the threshold using a precision-recall curve can minimize False Positives, both classes are equally significant. Metrics like MCC, Kappa, and F1-Score account for both precision and recall.

DISCUSSION

In this study, a Random Forest machine learning model was developed to predict student performance using LMS interaction data from 486 undergraduate students at a local university. The dataset contained 23 features. Feature Engineering was used to create representative features, and feature selection involved correlation analysis and domain knowledge to retain impactful features. The data was split into 70% training and 30% testing sets. SMOTE was applied to balance the dataset, and Robust Scaler handled outliers. Various machine learning techniques, including RF, SVM, LR, KNN, and NB, were evaluated using metrics suitable for imbalanced data: MCC, Kappa, and F1-Score. The Random Forest model achieved an MCC of 66.42%, a Kappa of 64.94%, and an F1 Score of 66.67%. In contrast, the Naive Bayes model performed the worst, with an MCC of 7.40%, a Kappa of 2.13%, and an F1 Score of 9.70%. Compared to models from similar studies (Brahim, 2022; Zeineddine et al., 2021), the Random Forest model underperformed in classification accuracy and F1-score but showed similar consistency in performance compared to the other models.

CONCLUSIONS AND RECOMMENDATIONS

In conclusion, Learning Management Systems (LMS) have significantly improved education by providing accessibility, centralized information, enhanced communication, and cost and time savings. However, challenges remain in extracting actionable insights from LMS data and implementing a Random Forest predictive model. This study aimed to understand online student behavior to identify students who need additional support or challenges. Using LMS data can provide valuable behavioral insights for students, lecturers, and institutions. A more detailed analysis of LMS data is recommended for future research.

Random Forest offers a perspective on the relative significance of various features within the model. This information may prove beneficial in comprehending the underlying data associated with the model and in evaluating the significance of various features. The Random Forest algorithm provides enhanced insights into the model, enabling a deeper understanding of the data and the identification of the factors that influence the model's outcomes.

Recommendations for future work include exploring alternative imbalance treatment techniques and expanding the classification model. Utilizing a larger and more reliable dataset would likely enhance the accuracy and robustness of the model's predictions. Additionally, addressing the dataset's inherent bias towards "Good" student classifications through techniques like ADASYN or SMOTE-ENN could improve overall performance. Finally, transitioning to a multi-class classification system with categories such as fail, bad, good, and excellent would provide a more granular understanding of student performance and inform targeted interventions.

IMPLICATIONS

The study emphasizes the importance of robust datasets and effective data handling techniques like SMOTE to enhance machine learning model accuracy in predicting student performance. The Random Forest model showed promise, but its underperformance suggests the need to explore alternative algorithms. Using balanced, well-scaled data and advanced imbalance treatments, as well as considering multi-class classification, is essential for detailed analysis. Future research should leverage LMS data for deeper insights into student behavior to improve educational outcomes. In the Philippine setting, this approach can significantly address educational disparities and enhance the quality of education. Educational institutions can use these models to identify students who may need additional support, thereby reducing dropout rates and improving overall academic performance.

ACKNOWLEDGEMENT

Special thanks to my loved ones for the inspiration.

FUNDING

The study did not receive funding from any institution.

DECLARATIONS

Conflict of Interest

The researcher declares no conflict of interest in this study.

Informed Consent

Not applicable.

Ethics Approval

Not applicable.

REFERENCES

Aguirre, P. M., & Legaspi, C. E. I. (2020). Predictors of academic performance of public elementary school learners. *Philippine Social Science Journal*, *3*(2), 63-64. https://doi.org/10.52006/main.v3i2.267

Artstein, R., & Poesio, M. (2008). Inter-Coder Agreement for Computational Linguistics. *Computational Linguistics*, 34(4), 555–596. https://doi.org/10.1162/coli.07-034-r2

Ayulani, I. D., Yunawan, A. M., Prihutaminingsih, T., Sarwinda, D., Ardaneswari, G., & Handari, B. D. (2023). Tree-based ensemble methods and their applications for predicting students' academic performance. *International Journal on Advanced Science, Engineering and Information Technology*, *13*(3), 919–927. https://doi.org/10.18517/ijaseit.13.3.16880

Beer, C., & Lawson, C. (2016). The problem of student attrition in higher education: An alternative perspective. *Journal of Further and Higher Education*, *41*(6), 773–784. https://doi.org/10.1080/0309877x.2016.1177171

Bernardo, A. B. I., Cordel, M. O., Calleja, M. O., Teves, J. M. M., Yap, S. A., & Chua, U. C. (2023). Profiling LOW-PROFICIENCY SCIENCE STUDENTS IN THE Philippines using machine learning. *Humanities & Social Sciences Communications*, *10*(1). https://doi.org/10.1057/s41599-023-01705-y

Brahim, G. B. (2022). Predicting student performance from online engagement activities using novel statistical features. *Arabian Journal for Science and Engineering*, *47*(8), 10225–10243. https://doi.org/10.1007/s13369-021-06548-w

Breiman, L. (2001). Random Forests. *Machine Learning, 45*(1), 5–32. https://doi.org/10.1023/a:1010933404324

Buctot, D. B., Kim, N., & Kim, S. (2021). Personal Profiles, family environment, patterns of smartphone use, nomophobia, and smartphone addiction across low, average, and high perceived academic performance levels among high school students in the Philippines. *International Journal of Environmental Research and Public Health*, *18*(10), 5219. https://doi.org/10.3390/ijerph18105219

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, *20*(1), 37–46. https://doi.org/10.1177/001316446002000104

Conijn, R., Snijders, C., Kleingeld, A., & Matzat, U. (2017). Predicting student performance from LMS data: A comparison of 17 blended courses using Moodle LMS. *IEEE Transactions on Learning Technologies*, *10*(1), 17–29. https://doi.org/10.1109/tlt.2016.2616312

Gorodkin, J. (2004). Comparing two K-category assignments by a K-category correlation coefficient. *Computational Biology and Chemistry*, *28*(5–6), 367–374. https://doi.org/10.1016/j.compbiolchem.2004.09.006

Imran, M., Latif, S., Mehmood, D., & Shah, M. S. (2019). Student academic performance prediction using supervised learning techniques. *International Journal: Emerging Technologies in Learning*, *14*(14), 92. https://doi.org/10.3991/ijet.v14i14.10310

Kaur, H., Kaur, T., & Garg, R. (2023). A prediction model for online student academic performance using machine learning. *Informatica*, *47*(1). https://doi.org/10.31449/inf.v47i1.4297

Lopes, A. P. (2014). Learning management systems in higher education. *In EDULEARN14 Conference* (pp. 5360-5365). Proceedings of EDULEARN14 Conference- IATED Publications.

Macaso, K. M. J., & Dagohoy, R. G. (2022). Predictors of performance in mathematics of Science, Technology, and Engineering students of a public secondary school in the Philippines. *Journal of Social, Humanity, and Education*, *2*(4), 311–326. https://doi.org/10.35912/jshe.v2i4.990

Martínez-Martínez, A., Montoliu, R., Salinas, J. A., & Remolar, I. (2023). Predicting student performance with virtual resources interaction data. *In IFIP advances in information and communication technology* (pp. 463–474). https://doi.org/10.1007/978-3-031-34111-3_39

Matthews, B. (1975). Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica Et Biophysica Acta. Proteins and Proteomics*, *405*(2), 442–451. https://doi.org/10.1016/0005-2795(75)90109-9

Pedregosa, P., Varoquaux, V., Gramfort, G., Michel, M., Thirion, T., Grisel, G., Blondel, B., Prettenhofer, P., Weiss, W., Dubourg, D., Vanderplas, V., Passos, P., Cournapeau, C., Brucher, B., Perrot, P., & Duchesnay, D. (2011). Scikit-learn: Machine learning in

Python. *The Journal of Machine Learning Research*, 12, 2825–2830. https://doi.org/10.5555/1953048.2078195

Raga Jr, R., & Raga, J. (2017). Monitoring class activity and predicting student performance using Moodle action log data. *International Journal of Computing Sciences Research, 1*(3), 1-16. Retrieved from //stepacademic.net/ijcsr/article/view/33

Raja, R., & Nagasubramani, P. C. (2018). Impact of modern technology in education. *Journal of Applied and Advanced Research*, S33–S35. https://doi.org/10.21839/jaar.2018.v3is1.165

Santos, M. S. M. D., Durano, D. C., & Hortillosa, A. D. (2023). The development of a proposed learning management system for senior high schools in the Philippines. *International Journal of Information and Education Technology*, *13*(3), 430–438. https://doi.org/10.18178/ijiet.2023.13.3.1823

Sasaki, Y. (2007). The truth of the F-measure. ResearchGate. https://www.researchgate.net/publication/268185911_The_truth_of_the_F-measure

Shahiri, A. M., Husain, W., & Rashid, N. A. (2015). A review on predicting student's performance using data mining techniques. *Procedia Computer Science*, *72*, 414–422. https://doi.org/10.1016/j.procs.2015.12.157

Shirinkina, E. V. (2022). Methods of data mining and educational analytics. Sovremennoe Obrazovanie, 1, 51–67. https://doi.org/10.25136/2409-8736.2022.1.37582

Sundhari, S. S. (2011). A knowledge discovery using decision tree by Gini coefficient. *2011 International Conference on Business, Engineering and Industrial Applications*, 232–235. https://doi.org/10.1109/icbeia.2011.5994250

Toro, J. M., & Reischl, V. (2018, January 1). Learning Management Systems. *Pressbooks*. https://pressbooks.pub/edd7032017f2/chapter/2/

Ulfa, S., & Fatawi, I. (2021). Predicting factors that influence students' learning outcomes using learning analytics in an online learning environment. *International Journal: Emerging Technologies in Learning*, *16*(01), 4. https://doi.org/10.3991/ijet.v16i01.16325

Warnars, H. L. H. S., Fahrudin, A., & Utomo, W. H. (2020). Student performance prediction using simple additive weighting method. *IAES International Journal of Artificial Intelligence*, *9*(4), 630. https://doi.org/10.11591/ijai.v9.i4.pp630-637

Wibawa, B., Siregar, J. S., Asrorie, D. A., & Syakdiyah, H. (2021). Learning analytic and educational data mining for learning science and technology. *AIP Conference Proceedings*. https://doi.org/10.1063/5.0041844

Zeineddine, H., Braendle, U., & Farah, A. (2021). Enhancing prediction of student success: Automated machine learning approach. *Computers & Electrical Engineering*, *89*, 106903. https://doi.org/10.1016/j.compeleceng.2020.106903

Zhang, Y., Fei, Q., Quddus, M., & Davis, C. (2014). An examination of the impact of early intervention on learning outcomes of at-risk students. *Research in Higher Education Journal*, 26.

Author’s Biography

Ali A. Naim is an Assistant Professor at the College of Computing and Information Sciences at the University of Makati. Mr. Naim is a graduate of Computer Science at Adamson University and a Master's in Information Systems at the University of Makati. He is presently finishing his PhD at Colegio de San Juan de Letran- Calamba.