RENAISSANCE
ISSN Number: 2582-8495
Bi-Annual, Multidisciplinary,
Peer Reviewed and Refereed E-Journal

ORIGINAL RESEARCH PAPER | COMMERCE | Vol. 1, June, 2025

STUDENT PERFORMANCE PREDICTION IN CS1 PROGRAMMING COURSES: A COMPREHENSIVE REVIEW OF MACHINE LEARNING TECHNIQUES

Ms. Hetal Savla*

ABSTRACT

Predicting student performance in CS1 programming courses is crucial for improving retention and learning outcomes. This review explores machine learning (ML) techniques applied to programming exercise datasets, which include submission logs, debugging behavior, and code quality metrics. By analyzing thirteen studies, the paper highlights trends such as early prediction models, explainable AI (XAI) frameworks, and adaptive learning systems. Challenges related to dataset diversity, real-time implementation, and scalability are discussed. The study provides insights into ML-driven interventions, emphasizing the role of programming exercise data in improving predictive accuracy and educational support systems.

Key words: Machine learning, CS1 courses, educational data mining, student performance prediction, programming education, explainable AI, adaptive learning

INTRODUCTION

Introductory programming courses, often referred to as CS1, are an inevitable part of almost every computer science curriculum, but they suffer immensely due to high failure and dropout rates. The earlier we are able to estimate students' performance, the more potentially helpful educators can be in helping students achieve better learning outcomes. Earlier, it was difficult to assess performance, but ML models are gaining more popularity because they can automatically analyse code and track changes in

*In-charge Principal & Assistant Professor – SMT C Z M GOSRANI BCA COLLEGE JAMNAGAR.

behavioural patterns such as submission and debugging without taking into consideration possibly skewed demographic data.

This study analyses ML-based approaches to student performance prediction using different types of techniques as posed in these research questions:

RQ1: Which ML algorithms are more effective for performance prediction in CS1 courses?

RQ2: How do explainable and adaptive ML models improve educational interventions?

RQ3: What are the challenges and opportunities provided by the scaling of ML models to a wider audience?

LITERATURE REVIEW

ML TECHNIQUES FOR PERFORMANCE PREDICTION:

(Llanos et al., 2023) have shown that Random Forest and SVM models can be used very effectively in early prediction through attempts at coding. Also, (Cooper & Juan, 2022) scoped out XAI frameworks from the perspective of diagnosing at-risk learners, which can help teachers make sense of predictions from models. (Albreiki, 2022) proposed a use case of a hybrid XAI framework in which ML and rules-based systems are used together for suggesting remedial measures, achieving a level of accuracy and understandability.

(Zhidkikh et al., 2024) showed the expectation of reproducible implementive predictive XAI in different scenarios. (Sunday et al., 2020) did student performance analysis through classification like Decision Trees and Naïve Bayes for effective categorization. Meanwhile, (Van Petegem et al., 2023) worked on the pre-submission debugging and submission behaviour and trends and showed that those were indeed important to model.

ADAPTIVE LEARNING AND HUMAN-AI COLLABORATION:

Both adaptive learning tools and collaboration AI are widely used to enhance the CS1 learning experience. For example, (Pereira et al., 2023) created a recommender system to assist the instructors with the assignment preparation. (Zhao et al., 2021) created a new framework ProLog2vec that is used by deep learning programmers for logging the embeddings of vectors to analyse the difficulties faced by novice programmers. (Mosquera et al., 2023) proved that the combination of the flipped classroom strategy with automatic code evaluation systems led to more engagement of students and improved outcomes.

(Cabo, 2021) discussed the iterative approaches for programming practices in ways that enable to predict student performance by focusing on loops and methods as critical ones. (Chopra et al., 2023) studied the theme's changes/discussion for the genre of online learning and how engagement trends may vary over them, providing in-depth information. (Islam et al., 2024) studied the performance of student academia over the internet using ML and emphasized on submission behaviour as key facet for academic achievement.

METHODOLOGY

RESEARCH QUESTIONS AND OBJECTIVES:

Three main research questions are examined in this study through the use of programming exercise datasets. The first investigates the best machine learning (ML) techniques for forecasting student success based on data unique to programming. The second looks at how interpretable findings from explainable and adaptive machine learning models might improve educational interventions. The study concludes by outlining the opportunities and difficulties associated with scaling predictive machine learning models for real-time applications using these datasets.

DATA COLLECTION:

The research methodology involved a systematic data assessment through study retrieval from the IEEE Xplore and Scopus and Google Scholar search phrases "programming exercise datasets", "student performance prediction". In order to be included in studies, the subject had to use programming exercise datasets including submission logs, as well as debugging behaviour, and achieve peer reviewed status, using quantitative metrics, such as accuracy, precision, recall, F1-score, and so on, from the time period between 2015 to the present. The research excluded investigations which lacked either programing-specific datasets or quantifiable outcomes. A performance assessment of different machine learning models based on programming exercise datasets evaluated their implementation outcomes while studying data qualities as well as ML techniques. Researchers categorised their findings into four distinct fields that included low prediction models alongside human-readable programming systems along with flexible education protocols and programme performance scalability limitations. Research on comparable algorithms determined which algorithms performed best alongside their data compatibility limitations. Research reveals programming data challenges due to under representative diversity and inability to extend data across real-time usage requirements. The research discoveries provide important educational implications together with valuable information that can guide future investigations even though standardized datasets are absent.

RESULTS

This study analysed thirteen peer-reviewed research publications to determine how machine learning methodologies could predict outcomes for students taking CS1 programming courses. The evaluation intensely focuses on programming exercise datasets that include metrics from code quality alongside student debugging activities and collected submission records. These datasets were essential for making precise predictions and offering useful information for educational initiatives.

The reviewed studies employed diverse ML algorithms, programming exercise datasets, and evaluation metrics to measure prediction effectiveness. Table 1 summarizes the findings:

TABLE 1: DATASET, ALGORITHM(S), METRIC(S), AND PERFORMANCE

Reference	Dataset	Algorithm(s)	Metric(s)	Value (s)
(Albreiki, 2022)	Programming exercise datasets with explainable ML and rule-based data	Hybrid ML + Rule-Based	F1-Score	82%
(Cabo, 2021)	Iterative programming practice data	Random Forest, SVM	Accuracy	86%
(Chopra et al., 2023)	Online discussion forums	Semantic Topic Chains	Engagement	82%
(Cooper & Juan, 2022)	Introductory programming course data	XGBoost, Explainable AI	Precision	85%
(Islam et al., 2024)	Online academic performance data	Logistic Regression	Accuracy	84%
(Llanos et al., 2023)	Early submission data from programming exercises	Random Forest, SVM	Accuracy	87%
(Mosquera et al., 2023)	Flipped classroom with auto-evaluation	Multi-layer Perceptron	Precision	84%
(Pereira et	Programming student	Explainable AI	Precision	81%

al., 2021)	behaviour data	Models		
(Pereira et al., 2023)	Assignment preparation data	Collaborative Filtering	Accuracy	88%
(Sunday et al., 2020)	Classification data from programming	Decision Trees, Naïve Bayes	Precision	80%
(Van Petegem et al., 2023)	Debugging logs from programming exercises	Decision Trees, SVM	Recall	78%
(Zhao et al., 2021)	Programming log data (ProLog2vec)	Deep Learning (ProLog2vec)	F1-Score	79%
(Zhidkikh et al., 2024)	Multi-institution analytics data	Gradient Boosting	Accuracy	90%

Researchers face two main challenges: they depend on individual institution data which restricts the use of findings beyond those boundaries (Sunday et al., 2020); (Zhidkikh et al., 2024). The performance limitations of machine learning frameworks make it hard for different education systems to use them (Zhidkikh et al., 2024). The essential feature of real-time system integration needs further study within adaptive learning space (Van Petegem et al., 2023). Data diversity problems arise because educational ML systems use school-specific datasets which limit their application to other settings (Sunday et al., 2020); (Zhidkikh et al., 2024). ML frameworks have problems when they need to work well across many different learning settings according to (Zhidkikh et al., 2024). A limited number of studies such as (Van Petegem et al., 2023) show how real-time integration supports adaptive learning systems. Data about student performance metrics and assignment submissions make up 41.7% and 25.0% of the datasets used to forecast student success. Data from online discussions makes up 16.7% of inputs while tests and quizzes count for 8.3%. Behavioural monitoring systems contribute 8.3% and student records are the last at 4.2%. Studies that make predictions focus on student learning data and engagement statistics.

FUTURE DIRECTIONS

Research investigations into the future should integrate detailed programming exercise submissions to develop improved predictive models for measuring student performance. A thorough evaluation of coding patterns including code organisation and error patterns together with debugging practices and task completion times reveals important measures for tracking both student development and their learning obstacles. Predictive models achieve higher accuracy when additional data including assignment resubmissions and plagiarism detection methods are combined with programming exercise details at a granular level. By combining these student performance insights with behavioural logs and demographic data institutions can gain a thorough understanding of programming education outcomes. Using this method allows educators to create customised support strategies which enhances programming course success for students.

CONCLUSION

The assessment focuses on demonstrating data's essential function for predictive analysis of student performance while giving extra emphasis to performance scores and assignment hand-in data. Student performance analysis can reach new heights through leveraging detailed data from programming exercise submissions and behavioural logs and demographic information in addition to existing primary data types. The field demands exploration of multiple data sources to improve model predictive capabilities and universal application. Explainable AI methods combined with cross-context model validation will establish efficient and equal solutions for predictive learning analytics. Improving predictive learning analytics standards will enable better educational support and better student outcomes.

REFERENCES

- Albreiki, B. (2022). Framework for automatically suggesting remedial actions to help students at risk based on explainable ML and rule-based models. International Journal of Educational Technology in Higher Education, 19(1), 1–26. https://doi.org/10.1186/S41239-022-00354-6/TABLES/9
- Cabo, C. (2021). Use of Machine Learning to Identify Predictors of Student Performance in Writing Viable Computer Programs with Repetition Loops and Methods. Proceedings -
- Frontiers in Education Conference, FIE, 2021-October. https://doi.org/10.1109/FIE49875.2021.9637302
- Chopra, H., Lin, Y., Samadi, M. A., Cavazos, J. G., Yu, R., Jaquay, S., & Nixon, N. (2023). Semantic Topic Chains for Modeling Temporality of Themes in Online Student Discussion Forums. Proceedings of the 16th International Conference on Educational Data Mining (EDM ’23), 67–78.

https://doi.org/10.5281/ZENODO.8115691

- Cooper, C. I., & Juan, S. (2022). Using Machine Learning to Identify At-risk Students in an Introductory Programming Course at a Two-year Public College. Advances in Artificial Intelligence and Machine Learning; Research. https://doi.org/10.21203/RS.3.RS-1025335/V7
- Islam, A., Bukhari, F., Sattar, M. A., & Kashif, A. (2024). DETERMINING STUDENT'S ONLINE ACADEMIC PERFORMANCE USING MACHINE LEARNING TECHNIQUES. Informatyka, Automatyka, Pomiary w Gospodarce i Ochronie Srodowiska, 14(3), 109–117. https://doi.org/10.35784/IAPGOS.6173
- Llanos, J., Bucheli, V. A., & Restrepo-Calle, F. (2023). Early prediction of student performance in CS1 programming courses. PeerJ Computer Science, 9, e1655. https://doi.org/10.7717/PEERJ-CS.1655/SUPP-2
- Mosquera, J. M. L., Suarez, C. G. H., & Guerrero, V. A. B. (2023). Effect of flipped classroom and automatic source code evaluation in a CS1 programming course according to the Kirkpatrick evaluation model. Education and Information Technologies, 28(10), 13235–13252. https://doi.org/10.1007/S10639-023-11678-9/FIGURES/8
- Pereira, F. D., Fonseca, S. C., Oliveira, E. H. T., Cristea, A. I., Bellhäuser, H., Rodrigues, L., Oliveira, D. B. F., Isotani, S., & Carvalho, L. S. G. (2021). Explaining Individual and Collective Programming Students' Behavior by Interpreting a Black-Box Predictive Model. IEEE Access, 9, 117097–117119. https://doi.org/10.1109/ACCESS.2021.3105956
- Pereira, F. D., Rodrigues, L., Henklain, M. H. O., Freitas, H., Oliveira, D. F., Cristea, A. I., Carvalho, L., Isotani, S., Benedict, A., Dorodchi, M., & De Oliveira, E. H. T. (2023). Toward Human-AI Collaboration: A Recommender System to Support CS1 Instructors to Select Problems for Assignments and Exams. IEEE Transactions on Learning Technologies, 16(3), 457–472. https://doi.org/10.1109/TLT.2022.3224121
- Sunday, K., Ocheja, P., Hussain, S., Oyelere, S. S., Balogun, O. S., & Agbo, F. J. (2020). Analyzing Student Performance in Programming Education Using Classification Techniques. International Journal of Emerging Technologies in Learning (IJET), 15(02), 127–144. https://doi.org/10.3991/IJET.V15I02.11527

- Van Petegem, C., Deconinck, L., Mourisse, D., Maertens, R., Strijbol, N., Dhoedt, B., De Wever, B., Dawyndt, P., & Mesuere, B. (2023). Pass/Fail Prediction in Programming Courses. Journal of Educational Computing Research, 61(1), 68–95. https://doi.org/10.1177/07356331221085595/ASSET/IMAGES/LARGE/10.1177_07356331221085595-FIG13.JPEG
- Zhao, H., Li, M., Lin, T., Wang, R., & Wu, Z. (2021). ProLog2vec: Detecting Novices' Difficulty in Programming Using Deep Learning. IEEE Access, 9, 53243–53254. https://doi.org/10.1109/ACCESS.2021.3067505
- Zhidkikh, D., Heilala, V., Van Petegem, C., Dawyndt, P., Järvinen, M., Viitanen, S., De Wever, B., Mesuere, B., Lappalainen, V., Kettunen, L., & Hämäläinen, R. (2024). Reproducing Predictive Learning Analytics in CS1: Toward Generalizable and Explainable Models for Enhancing Student Retention. Journal of Learning Analytics, 11(1), 132–150. https://doi.org/10.18608/jla.2024.7979