IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 17 November 2022, accepted 4 December 2022, date of publication 6 December 2022,
date of current version 19 December 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3227072

RESEARCH ARTICLE

Earliest Possible Global and Local Interpretation of Students' Performance in Virtual Learning Environment by Leveraging Explainable AI

MUHAMMAD ADNAN[1], M. IRFAN UDDIN[1], EMEL KHAN[2], FAHD S. ALHARITHI[3], SAMINA AMIN[1], AND AHMAD A. ALZAHRANI[4]

[1]Institute of Computing, Kohat University of Science and Technology (KUST), Kohat 26000, Pakistan
[2]Institute of Numerical Sciences, Kohat University of Science and Technology (KUST), Kohat 26000, Pakistan
[3]Department of Computer Science, College of Computers and Information Technology, Taif University, Taif 21944, Saudi Arabia
[4]Department of Information Systems, College of Computers and Information Systems, Umm Al-Qura University, Mecca 24382, Saudi Arabia

Corresponding author: M. Irfan Uddin (irfanuddin@kust.edu.pk)

This work was supported by the Taif University Researchers Supporting Project, Taif Unversity, Taif, Saudi Arabia, under Grant TURSP-2020/347.

ABSTRACT In this research study, we propose an Explainable Artificial Intelligence (XAI) model that provides the earliest possible global and local interpretation of students' performance at various stages of course length. Global and local interpretation is provided in such a way that the prediction accuracy of a single local observation is close to the model's overall prediction accuracy. For the earliest possible understanding of student performance, local and global interpretation is provided at 20%, 40%, 60%, 80%, and 100% of course length. Machine Learning (ML) and Deep Learning (DL) which are subfields of Artificial Intelligence (AI) have recently emerged to assist all educational institution's in predicting the performance, engagement, and dropout rate of online students. Unfortunately, traditional ML and DL techniques lack in providing data analysis results in an understandable human way. Explainable AI (XAI), a new branch of AI, can be used in educational settings, specifically in VLEs, to provide the instructor with the study performance results of thousands or even millions of online students in a human-understandable way. Thus, unlike black box approaches such as traditional ML and DL techniques, XAI can help instructors to interpret the strengths and weaknesses of an individual student, providing them with timely personalized feedback and guidance. Various traditional and various ensemble ML algorithms were trained on demographic, clickstream, and assessment features to determine which algorithm gives the best performance result. The best-performing ML algorithm was ultimately selected and provided to the XAI model as an input for local and global interpretation of students' study behavior at various percentages of course length. We have used various XAI tools to give students' performance reports to instructors, in an explicable human way, at different stages of course length. The intermediate data analysis and performance reports will help instructors and all key stakeholders in decision-making and optimally facilitate online students.

INDEX TERMS Global explainability, local explainability, explainable AI, course length, decision making, artificial intelligence, personalized feedback, earliest possible intervention, earliest possible interpretation.

I. INTRODUCTION

In the last three decades, the emergence of the Internet has played a crucial role in the use of online learning platforms (distance learning, e-learning, Virtual Learning Environments (VLEs), mobile learning (M-learning)) [1]. In VLEs, there are no temporal or unique constraints; therefore, they encourage and favor those students to enroll who cannot afford to take physical classes. With the advent of Learning Management Systems (LMS), students are

The associate editor coordinating the review of this manuscript and approving it for publication was Tony Thomas.

provided with easy-to-use asynchronous and synchronous support tools such as course material in the form of videos, animations, audio, and text; messaging tools such as emails, chats, messaging systems, and reference tools such as wikis, forums, dictionaries, and problems solutions [2]. By mining the LMS logs, students' study behavior and performance in the enrolled course can be elicited and their interactions with the LMS can be analyzed.

In VLEs, students interaction includes the number of times the student logged into the system, learning time, learning duration, the number of times a particular course material has been accessed, online forum participation, interaction with the instructor in the form of messages, repetition rate, problem-solving rate, and the number of times a quiz was taken. Analyzing students' learning behavior is essential as it helps instructors provide tailored learning content, personalized feedback, and assistance at the optimal time, thus, keeping students on the right track. Providing timely feedback and personalized learning materials can also help reduce the number of students at risk of dropout or failure. Therefore, Educational Data Mining (EDM) can help all the stakeholders involved in online learning, such as students, administrators, instructors, and coordinators, make the right decisions at the right time.

Educational Data Mining (EDM) usually uses AI techniques and algorithms to train computers to understand the learning behavior of different students [3], [4], [5]. Online learning platforms can track every interaction of students with the registered course, thus providing abundant data for AI techniques to process and report on students' study behavior and ultimately improve their performance. AI techniques with the availability of historical interaction data can help instructors know students' learning behavior at various stages of course length, even at the beginning of the course, provided that student background and demographic information are available [6], [7]. Previous studies have proved that ML and DL, subfields of AI, can be used to analyze students' historical data and provide valuable insight [8], [9]. In general, these studies use ML and DL techniques to predict students' dropouts, success, failure, engagement intensity, answer correctness prediction, and performance [10], [11], [12], [13], [14]. In these studies, primarily, the prediction is performed at the end of the course length. The prediction results are then used to motivate and encourage students to improve their performance in the upcoming semester. The drawback of predicting the students' performance at the end of the semester is that students are not motivated in their current semester, which can result in students' early dropout. Few studies have been conducted that try to predict students' performance right from the start of the course length [15], [16]. Subsequently, the earliest possible intervention is possible, which can encourage students to stay on the right path. In addition to predicting students' performance, visualization techniques are now commonly used to observe students' learning behavior [17]. Numerical methods assist instructors in knowing about minor learning habits and can be used to unveil unknown hidden learning strengths and weaknesses [18]. Moreover, students can be classified into various groups according to their performance to provide adaptive and personalized learning content [19], [20].

Developing an XAI predictive model that can interpret and predict students' learning behavior as early as possible in the registered course is challenging. Creating an XAI predictive model that can identify students' at-risk of failure and explaining to the instructors the main causes of failure in an easy and human-understandable way can lead to developing a system that provides intelligent feedback and suitable action recommendations to support students in self-regulated studies. Creating an explainable AI model is supported by USA Defense Advanced Research Projects Agency (DARPA). XAI scientific challenge launched in 2016 stated that current AI systems; however, they have many benefits in different fields, but most lack in explaining their decisions to humans in a simple way [21]. When adequately developed and implemented, XAI systems promise to benefit people through explainability, interpretability, and transparency [22], [23]. Apart from education, other domains such as defense, health, finance, and law need XAI systems because it is crucial to understand the decisions and build trust in XAI systems [24], [25].

Currently, ML and DL techniques are used by researchers to make data-driven decision-making systems. But most ML/DL algorithms that are used today to extract information from the data mostly follow the black box approach [26]. Researchers and practitioners who know the hidden working mechanism of ML and DL techniques understand how they work and make decisions. However, ordinary people using these automated systems struggle to know how a particular decision is made and therefore are reluctant to trust AI-based automated systems [27]. Whether in education or any other sector, ordinary people need to explain how AI-based system develops, works, and makes decisions. Therefore, XAI models try to explain or justify how AI models make predictions. Moreover, once the internal working of the model is known, then the working methodologies of the model can be improved in the future for its performance improvement. Apart from the field of academia and online learning, the use, and applications of XAI are ubiquitous such as in the area of machine vision [28], machine hearing [29], natural language processing [30], robotics process automation [31], natural language generation [32], machine translation [33], speech synthesis [34], optical character recognition [35], handwriting recognition [36], image processing and recognition [37], facial recognition [38], health [39], self-driving cars [40], pattern recognition [41], and online fraud detection [42], etc.

Traditional ML models act like a black box where input is given in the form of features, and the models try to inspect or understand the steps taken while making decisions. For example, features associated with an online learner are provided and processed by an ML algorithm. Most of the time, these ML/DL algorithms work like a black box, and

a decision or prediction is made on the success or failure of an online learner in the future. The decision, in this case, is binary, and the algorithm just outputs whether the student will be successful or unsuccessful. On the other hand, XAI models also provide reasons or explanations in a human-understandable way on why a specific student will be successful or unsuccessful. The reasoning or explanation power gives XAI several advantages over traditional ML approaches. XAI models encourage VLEs stakeholders to make crucial decisions without hesitation as the automated process is transparent and interpretive. In the future, instructors can tell students about the reasons based on which recommendations and feedback were provided to them. XAI models can also encourage instructors to provide targeted recommendations based on students' VLE interaction information and performance.

While deploying and implementing ML models, there is often a tradeoff between model accuracy and interpretability [43]. It has been noticed that complex models such as neural networks (Feed-Forward Neural Networks (FFNN), Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNN), and transformers) have high performance on large datasets and low interpretations [44]. On the other hand, simple models such as linear models, Decision Trees (DTs), and Support Vector Machines (SVMs) provide high interpretations about their predictions and face lower performance [45]. Therefore, the designers should know which ML and DL model to choose that is interpretive and has high performance. Generally, ensemble models such as Random Forest (RF), adaptive boosting, gradient boosting, and Extreme Gradient Boosting (XGB) show acceptable performance and interpretation [46].

There are numerous XAI toolsets and libraries with pros and cons, but researchers can use them according to their needs and depending on which ML/DL algorithm they use. Currently, popular XAI toolsets include Local Interpretable Model-agnostic Explanations (LIME) [47], Layer-Wise Relevance Propagation (LWRP) [48], and XEMP Prediction Explanations [49], DeepLIFT [50], and Shapley Additive exPlanations (SHAP) [51]. LIME targets DL and supervised ML models in their current state. It can provide an acceptable explanation for any given supervised ML model by separately treating it as a black box. LWRP is one of the most protuberant and prominent frameworks used in XAI. LWRP targets layered neural networks such as CNN, RNN, Artificial Neural Networks (ANNs), and LSTMs. For example, if a neural network envisages cancer identification from a mammogram, then the description given by LWRP would be a map of which picture element in the original image contribute to the judgment and what magnitude.

XEMP-based XAI toolsets differ from others in their ability to generate prediction explanations for multi-class classification problems. The disadvantage of using XEMP-based toolsets is that computing prediction explanations classification task is resource intensive. The main components of XEMP-based toolsets include computation inputs, prediction threshold values, prediction explanations preview, and calculators to compute explanations for fully designated predictions. Deep Learning Important FeaTures (DeepLIFT) mainly uses reference activation and compares the activation of each neuron to it. Furthermore, a contribution score is assigned to each neuron according to how much there is a difference between each neuron's activation value and reference activation value. DeepLIFT methods can also divulge necessary dependencies and features that other XAI methods could not provide. As the name suggests, DeepLIFT mainly targets interpreting deep neural network models such as ANN, CNN, RNN, LSTM, and transformers. SHAP open source library, developed by Microsoft, is implemented to explain the working of the ML/DL models using shapely values. SHAP can primarily explain ensemble models such as tree ensembles using an API called TreeSHAP.

A DL models explanations can also be provided using an API called deepSHAP. In a scenario where it is unknown what form of the algorithm a model is using, especially for a model-agnostic explanation, a toolset called KernelSHAP can be used. Therefore, the SHAP library can target linear, tree, DL, and multi-stage combinations of models such as transformers and LSTM. The concepts used by SHAP for model explanations are inherited by the game theory, mainly composed of two components, i.e., a game and some players. The players act like features provided to the model, and the game is responsible for producing the model's outcomes. While using SHAP, the importance of each player is determined by shapely values, which are based on the idea that the outcome of each possible coalition of players should be considered to assess the impact of each player on the output values.

Some other objectives of this research work include:

- To predict the students' performance at various percentages of course length.
- To determine the features the ML model thinks are important and impact the overall decision.
- Local explainability: How is a particular prediction by the model affected by each feature?
- Global explainability: How is each feature's prediction affected by a generalized ML model?
- What is the effect of each feature when a larger number of predictions are considered?

Moreover, the study will facilitate research and data scientists to perform debugging tasks quickly, build trust, oversee future data collection, and help instructors make the right decision.

The rest of the paper is organized into various sections. Section II discusses previous studies related to the application of machine learning in predicting students' performance i.e., predicting at-risk students, engagement predictions, predicting performance at the end of the course, and earliest possible performance prediction. Section III describes the dataset used in this research study. Section IV is about the various experiments carried out for the earliest possible prediction and interpretation of online students' study

behavior. Section V concludes this research study along with its limitations and future work.

II. BACKGROUND AND RELATED STUDIES

This section analyzes the previous studies that were carried out in the area of Artificial Intelligence in Education (AIE), EDM, XAI in education, and Learning Analytics (LA). The objective is to study how AI, ML/DL, and XAI techniques were used in determining the learning behavior of online students and what measures were taken to improve their performance. This section is further divided into different sections according to various studies carried out in determining students' online engagement, students' dropout, students' performance prediction, and next answer correctness prediction while using XAI and ML/DL techniques.

A. STUDENTS' ONLINE ENGAGEMENT PREDICTION

Using online logging data and clickstreams to gain insight into the learning engagements of online students is a vital and challenging task. Knowing about the earlier learning engagements leads to designing a compelling and actionable predictive model that could be used for timely intervention. In [52], the authors extracted important learning features from students' interaction data to determine their engagement intensity. Based on these features, the TrAdaBoost-based transfer learning model was proposed. The model was trained on previous course interaction features and was used in the current study semester to determine the model's generalization ability and predict new students' engagement behavior. The experimental results revealed that the model achieved high precision and accuracy even when the recent data was insufficient to train the model. Moreover, the model effectively assisted instructors in helping students at risk of dropout and failure.

In VLE, it is essential to distinguish between course completers and non-completers for tailored and relevant recommendations and feedback [5]. The difference between the two groups can be revealed by examining their engagement features in their logins, logouts, clicks, time duration, study time, preferences, etc. A learning analytics method was used in [53] to examine four online courses with identical pedagogical models. In all 13 considered features, the study results revealed a significant difference between the online engagements of students who completed the course and those who did not. Successful students' engagement intensity was twice as high as unsuccessful students except for posting problems on online forums. The study proves that success in the final examination is directly related to students' online engagement in various activities of the online registered course.

A significant problem that online learning environments (such as Coursera, udemy, udacity, Edx, etc.,) face is the retention of students once they have registered for a particular course. Research studies reveal that the reason behind discontinuing an online course is that students primarily take courses for skills improvement and not for getting completion certificates. Therefore, students leave that course when a problem is solved, or a skill is mastered. It has been observed that dropout is the most concerning factor in the continuity of an online course. Educators and researchers have studied the significant reasons behind students' dropout by analyzing their academic information and online learning behavior. Subsequently, various learning models and strategies have been proposed to reduce students' dropouts and improve their study behavior. A study carried out by [10] noticed that dropout prediction is a time-series problem, which needs students' continuous modeling daily, hourly, or even every minute. The proposed model integrated the regularization term into a logistic regression model. The other proposed model was the Input-Output Hidden Markov Model (IOHMM), which achieved an accuracy of 84% in predicting students at risk of dropout compared to the baseline ML/DL models.

In another interesting work, A. Kaur et al. [54] carried out a study in which students' online engagement was extrapolated from their facial expressions, such as body movements, gaze patterns, and facial expressions. The variations in students' engagement were recorded, and various features were extracted to reveal students' behavior while they were watching educational videos. Subsequently, students' engagement level was associated with subject behavior features, and different output labels annotated the features. A deep multiple-instance learning framework was proposed to detect online students' engagement intensity at various stages of video length. The framework can then be used by VLEs and Massive Open Online Courses (MOOCs) to design course video material.

B. STUDENTS' PERFORMANCE PREDICTION

Various studies have been carried out that predict students' online performance in two ways, i.e., predicting students' performance at the end of the course and the earliest possible prediction of students' performance in the registered class. The following section discusses studies related to both practices.

1) STUDENTS' PERFORMANCE PREDICTION AT THE END OF THE COURSE

Most studies that leverage ML/DL techniques predict students' performance at the end of the course length [20], [55]. There are advantages and disadvantages to predicting students' performance at the end of the course. One main advantage of using ML/DL techniques to predict students' performance at the end of the course length is that ML/DL algorithms are provided with enough data to train them and to make them more generalizable. At the end of the course length, there is enough data about online students' interactions which ML/DL algorithms can use to determine the strength and weaknesses of students during their study. A trained and generalizable model is then ready to be tested on the same students in the next course or on new students

in the same course. The disadvantage of predicting students' performance at the end of the course length is that instructors are unable to perform the earliest possible performance prediction in the current course for needed support and feedback. Due to a lack of proper feedback, students may drop out earlier in the course.

Ghorbani and Ghousi [19] compared various resampling techniques such as Random Over Sampler, SMOTE-Tomek, SVM-SMOTE, SMOTE-ENN, and Borderline SMOTE to predict students' performance using two different datasets while also handling imbalanced data problems. Additionally, various ML/DL algorithms such as Naïve Bayes, Logistic Regression, Decision Trees, SVMs, XG Boost, and ANNs were used to check which resampling technique shows better performance. The results revealed that the model trained using nominal features and fewer classes for classification will generate better results. Moreover, the model delivers better results when trained on a balanced dataset than a model trained on an imbalanced dataset. When conducted, the Friedman test confirmed that SVM-SMOTE is an efficient resampling method, and the Random Forest (RF) model achieved the best results compared to other models.

Most research studies used supervised ML/DL techniques to create learning models and to study students' characteristics inducing their performance and preferences. The reason for using supervised ML/DL techniques in eliciting students' performance is due to the nature of their learning features. Independent variables include study time, duration, preferences, number of logins/logouts, online participation, and preferred learning material. In contrast, students' final performance is a dependent variable that supervised ML/DL algorithms try to predict. Due to the interrelation between independent and dependent features, supervised types of ML/DL techniques are used in EDM. Besides the supervised ML/DL techniques, numerous studies have been carried out that use unsupervised and semi-supervised ML/DL methods to predict students' performance at the end of their final examinations. A research study carried out by [56] examined and evaluated two wrapper methods in conjunction with semi-supervised methods for predicting students' performance at the end of the course length. The study showed that semi-supervised ML/DL techniques could be utilized to create a trustworthy predictive model. Moreover, classification accuracy and precision can significantly be improved by using fewer label features and many unlabeled features. Finally, more accurate supervised models can be trained on the already clustered data by semi-supervised or unsupervised ML/DL methods.

X. Xu et al., [57] highlighted some key factors that can be considered to know how students' academic performance can be predicted and differentiated from Internet usage behavior. Moreover, some new metrics were proposed that can be utilized to evaluate and assess students' academic performance. The study showed that behavior discipline plays a pivotal role in students' academic success, and the prediction accuracy of the ML model can be increased by adding more features. Internet-connection frequency variables are positively associated with academic performance, whereas Internet traffic intensity variables are adversely related to academic achievement.

During the COVID-19 pandemic, remote learning was widely adopted at all education levels, especially at the university level. The sudden adaptation to the new learning environment initiated many hidden and unseen problems for online students. In a short time, it was difficult for the VLEs stakeholders to understand the factors that impact student performance. Ho IM et. al. investigated important features that influence the performance and satisfaction of undergraduate students who have adopted emergency remote learning while using Microsoft Team and Moodle as key learning means [58]. Using the RF recursive features elimination process, a comparison between various ML models and multiple regression models was made, considering predictive accuracy as a key metric. The results showed improved accuracy in all ML and all multiple regression models, with the elastic net regression model being the most accurate one with 65.2% explained variance.

2) EARLIEST POSSIBLE PERFORMANCE PREDICTION IN THE CURRENT SEMESTER

Although there are numerous advantages of VLEs platforms, they also face critical challenges such as developing self-regulated learning behavior, low engagement, low motivation, high dropouts, and forcing students to set their own goals. A study conducted in [6] aimed to predict the earliest possible performance of online students' by dividing the course length into six parts. The student's performance was predicted at 0%, 20%, 40%, 60%, 80%, and 100% of course completion, thus facilitating instructors to perform a timely intervention to avoid student early dropouts. The study showed that time-dependent features, engagement intensity in the form of click stream data, and assessment scores were significant factors in determining students' online behavior. When trained using the RF algorithm, the predictive model gave the best score regarding accuracy, recall, precision, and F-score.

Another research study carried out by [59] utilized various ML techniques to predict and identify possible failing students early in the course, i.e., at week 4 of the semester. ML models achieved an accuracy of 97.2% for pass-fail students and 88.0% for failure mode matches. The results showed that the earliest identification of struggling students is possible, and ML techniques can be used in an applicable pedagogical context to support their use in a complete student support system.

The earliest possible performance prediction and students' classification are helpful in online learning environments. It enables university administrators and instructors to manage resources and properly help students achieve good results [43]. The most prominent problem researchers faced in determining the earliest possible performance prediction of online students is the lack of big data associated with

VLEs in students' interactions with the online system [60]. But recently, several online learning platforms have made their data public and anonymous for researchers to help them identify key learning factors that significantly impact students' learning behavior [61]. With the growing availability of large datasets associated with online learning platforms, early students' performance prediction has become popular and necessary in recent years.

Moreover, Learning Management Systems LMS can be used for logging students' activity data in most academic institutions. A research study conducted by [62] leveraged deep learning neural networks called LSTM networks to analyze students' online temporal study behavior. Temporal study behavior relates to analyzing how students perform every second or every minute. Such problems are also called time-series problems. The study results indicated that LSTM networks are very good at identifying students' time-series behavior compared to conventional ML models. Time series data such as students' clickstreams successfully facilitated LSTM networks for the earliest possible detection of students at risk of failure or dropout. Additionally, DL models have stronger generalizability and higher performance scores in time-series-related problems than traditional ML algorithms.

In other related work, D. Baneres et al. [63] proposed an early warning system. It displayed the students' states through dashboard visualization for students and teachers. Subsequently, an early feedback prediction system was developed to help instructors to perform personalized interventions, thus reducing the risk of students' early dropouts. When evaluated, the early warning system successfully identified students at risk of failure with acceptable accuracy and spotted the most common features that trigger dropouts.

Continuous research and advances in ILS, LMS, VLE, and MOOCs promise to develop and produce autonomous learning systems that will learn, think, decide, act, and interfere independently. However, one significant inability of the studies mentioned above is that current ML/DL techniques are limited by their inherent implementation and methodologies to explain their working, decision-making, and action to humans in a simple and understandable way. Explainable AI (XAI) techniques, technologies, and associated tools promise to make ML/DL techniques understandable, trustworthy, and manageable for ordinary humans. A study related to developing an interpretable model by utilizing explainable AI was carried out by Kostopoulos et. al. [64]. In the study, an interpretable model was created for the earliest possible prediction of MOOCs certificate completion. The results revealed that Light Gradient Boosted Machine, Logistic Regression, and Gradient Boosting models showed the best results in terms of accuracy, AUC curve, recall, precision, F1-score, and Kappa and Matthews correlation coefficient. Another study was carried out by Alwarthan et al. [65] in which an explainable AI model was developed for the identification of students who are at risk of failure in higher education. The SMOTE-Tomek Link technique was utilized for balancing the three imbalanced datasets. Finally, LIME and SHAP explainable AI techniques were used to interpret and explain the proposed ML models.

A study related to explainable AI was conducted by Stamatis K. et al. which utilized a semi-regression algorithm for predicting and interpreting the grades of undergraduate students in their final examination in one year course [66]. By utilizing various explainable AI methods, the features that contributed the most to improving the final performance were interpreted and analyzed. The experimental results showed that semi-supervised techniques as compared to supervised ML techniques can do a better job in the earliest possible identification of students who are at risk of failure.

In this research study, our main objective is to create an explainable and predictable ML model (EPMLM) AI (XAI) model that can describe how students learning behavior is modeled and how the ML model makes various decisions. XAI model will help instructors to make timely interventions and provide feedback to students in a responsible way. To build instructors' confidence in VLEs, the instructors need to retrace and comprehend how the VLE has predicted the performance of a particular student. The online learning platforms integrated with AI methods perform the whole process using a black-box approach that is almost impossible to interpret. XAI model will assist administrators and instructors in answering important questions like why a particular student is at-risk of failure from the start of the semester, why a student has a low level of engagement, what essential features play a significant role in student learning, why a student was intervened and persuaded for improving their performance, and more importantly XAI model will build the trust of instructors in how it has made a particular decision.

III. DATASET DESCRIPTION

For determining the earliest possible interpretation of students' study behavior and performance, a freely accessible dataset available at `https://analyse.kmi.open.ac.uk/open_dataset`, provided by Open University, UK, and certified by Open Data Institute `http://theodi.org/`, was utilized. The dataset consists of students centered data such as students' online interactions, students' assessments scores, registration information, students' demographics, course information, and students' clickstreams. The data is spread across 7 tables representing various entities and are connected through key identifiers. Students' interactions with the VLE are stored in the form of clickstream data in the student VLE table, whereas information about students' assessments scores is stored student assessment table. The dataset contains information about 7 courses and 22 modules with 32,593 registered online students. The students' demographics include students' ID, gender, immigration band, highest education, age band, number of previous attempts, credit hours already studied, disability, region, and final score. Throughout the course, students submit various assessments related to each course module and are evaluated by assessment scores. Table 1

TABLE 1. Features along with their descriptions used for training various ML models. The important features included assessment scores, clickstream statistics, and demographic features.

Features	Description
Assessment Score	Student score in an assessment i.e., quiz, assignment, etc.
Weighted Cumulative Score (CS)	Weighted Cumulative Score at 20%, 40%, 60%, 80%, and 100% of course length.
Percentage Weighted Cumulative Score (PCS)	Percentage Weighted Cumulative Score at 20%, 40%, 60%, 80%, and 100% of course length.
Late Assessment submission (LA)	Late Assessment submission (LA) Score at 20%, 40%, 60%, 80%, and 100% of course length.
Average of the assessment the raw score (RS)	Average of the assessment raw score (RS) at 20%, 40%, 60%, 80%, and 100% of course length.
The sum of clicks per course module (SC),	The sum of clicks per course module (SC) at 20%, 40%, 60%, 80%, and 100% of course length.
Average clicks per course module (AC)	Average clicks per course module (AC) at 20%, 40%, 60%, 80%, and 100% of course length.
Mean Clicks per course module	Mean Clicks per course module (MC) at 20%, 40%, 60%, 80%, and 100% of course length.
Student Id	For student unique identification.
Gender	Whether a student is male or female
Immigration Band	Indicates the depravity band of the area where the student stayed for the duration of the module presentation
Highest Education	The highest education the student has before registering for a course.
Age Band	Indicates the student's age
Number of Previous attempts	The number of times the student has taken this course
Studied credits	The total number of credits the student has taken before taking the current module representation.
Region	To which area student belongs
Disability	Indicates whether the student has a disability or not.
Final Score	The final score in the registered course.

hlpresents the features along with their descriptions used for modeling various ML algorithms.

A. DATA PREPROCESSING

For the earliest possible interpretation of students' study behavior and the creation of efficient ML models, all missing values, outliers, and noise data were either removed or replaced by their average value. As students' performance was evaluated at various stages of course length, it was ensured that essential features such as assessments date had no invalid information, and the mean values replaced the missing dates.

B. FEATURE ENGINEERING

We extracted some more features from the existing features to show students' interaction activities to instructors in a simple and human-understandable way. These features were extracted at 20%, 40%, 60%, 80%, and 100% of course length. The features included Weighted Cumulative Score (CS), Percentage Weighted Cumulative Score (PCS), Late Assessment submission (LA), the average of the assessment Raw Score (RS), the sum of clicks per course module (SC), Average clicks per course module (AC). We also predicted the students' performance by using only demographic features. To summarize, students' performance was determined and predicted using only demographic features, 20%, 40%, 60%, 80%, and 100% course completion data. This way, it would be easier for instructors to investigate the insight of students' study behavior right from the start of the course and at various lengths. The new extracted features included Weighted Cumulative Scores (CS20, CS40, CS60, CS80, CS100), Percentage Weighted Cumulative Score (PCS20, PCS40, PCS60, PCS80, PCS100), Late Assessment submission (LA20, LA40, LA60, LA80, LA100), Assessment Raw Score (RS20, RS40, RS60, RS80, RS100), Sum of Clicks per course module (SC20, SC40, SC60, SC80, SC100), and Mean Clicks per course module (MC20, MC40, MC60, MC80, MC100). More information about these features is presented in table 1.

IV. METHODOLOGY

The workflow diagram in figure 1 shows the different phases of the methodology. In phase 1, six traditional ML models were utilized to predict students' performance at various stages of course lengths. The six traditional ML models included logistic regression, Stochastic Gradient Descent (SGD) classifier, gaussian Naïve Bayes (NB), K-Nearest Neighbor (KNN), Decision Tree (DT) classifier, and linear Support Vector Classifier (SVC). Training various traditional ML models determined which model gives the best results for predicting students' performance at different percentages of course lengths. The models were trained on all independent features, including demographic, clickstream, and assessment scores. The models were also trained after performing features merge operations where the Distinction and Pass classes were combined into the Pass class, and the Fail and Withdrawn classes were combined into the Fail class.

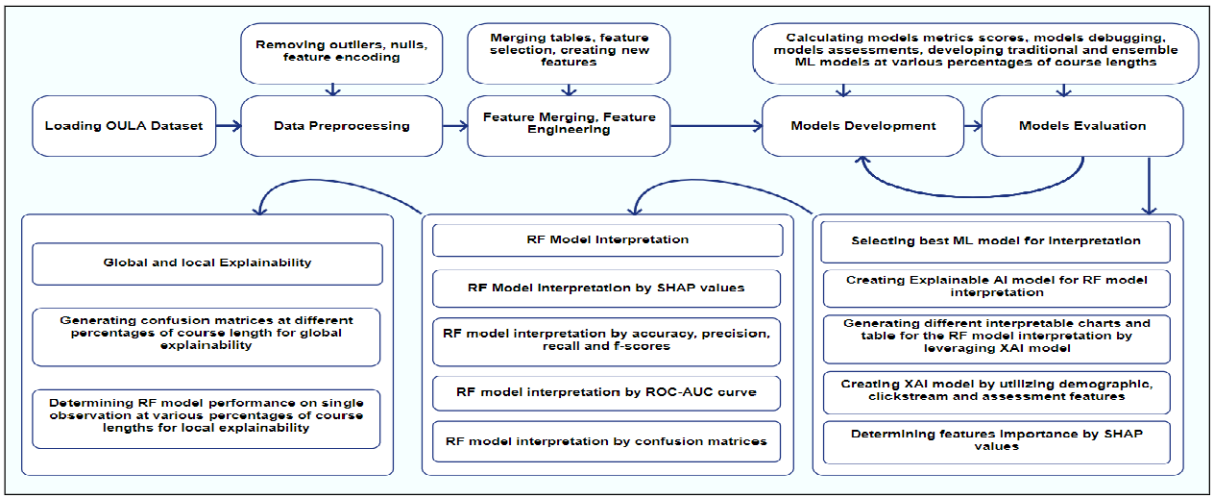


FIGURE 1. Earliest possible local and global interpretation of students' performance by utilizing the RF and XAI models.

For training and testing the various traditional ML models, the dataset was split into training and testing sets by an 80:20 percent ratio i.e., 80% data was used for training the models whereas 20% data was used for testing the models. Moreover, to avoid models suffering from the underfitting problem, the k-fold cross-validation technique was used with the value of k set to 10. Lastly, all the models were trained on 20%, 40%, 60%, 80%, and 100% course data.

In phase 2, we employed six ensemble ML models to predict students' performance and various percentages of course lengths. The six ensemble ML models included Bagging Classifier, Random Forest (RF) Classifier, Extra Tree Classifier, Gradient Boosting, Adaptive Boosting Classifier, and Voting Classifier. Similar to traditional ML models, the purpose of training various ensemble models was to determine which model gives the best results in terms of accuracy, precision, recall, and f-score at various percentages of course lengths. First, all six ensemble models were trained on all 45 independent features (features related to demographic, clickstream, and assessment). Secondly, the six ensemble models were also trained after the feature merge operation. Moreover, all ensemble models were trained on only demographic features, and on 20%, 40%, 60%, 80%, and 100% of course data. Lastly, the best multiclass classification algorithm is selected for local and global interpretation where the XAI model explains how a particular decision was made or how a specific prediction was performed.

In phase 3, we used explainable AI to perform the earliest possible interpretation of students' study behavior. Various explainable AI (XAI) tools and methods were used to interpret students' study behavior at different phases of course length. In phase 4, an XAI model was created using demographic and clickstream data. In phase 5, the XAI model was improved by incorporating students' assessment scores. Phase 6 discusses global explainability, where various confusion matrices were generated to explain the overall performance of the RF model. Phase 7 discusses local explainability, where the XAI model was created to delineate the performance of the RF model on a single observation at 20%, 40%, 60%, 80%, and 100% course length.

A. PHASE 1, BLACK BOX APPROACH: USING TRADITIONAL ML ALGORITHMS FOR PREDICTING STUDENTS' PERFORMANCE AT VARIOUS STAGES OF COURSE LENGTH

Before providing features to ML algorithms, some necessary preprocessing steps were performed. The students' demographic table was merged with the assessment table. The demographic table contained features such as code module, code presentation, student id, gender, region, highest education, immigration band, age band, number of previous attempts, studied credits, disability, and final result score. The assessment table contained features such as code_module', code_presentation, id_student, CS20, CS40, CS60, CS80, CS100, PCS20, PCS40, PCS60, PCS80, PCS100, LS20, LS40, LS60, LS80, LS100, RS20, RS40, RS60, RS80, RS100, date of registration. Furthermore, students' click stream information stored in the VLE table was also merged with the student demographic table using the left join operation. The VLE table consisted of the code module, code presentation, student id, sum clicks0, sum clicks20, sum clicks40, sum clicks60, sum clicks80, sum clicks100, mean clicks0, mean clicks20, mean clicks40, mean clicks60, mean clicks80, and mean clicks100. As mentioned earlier, the numbers 0, 20, 40, 60, 80, and 100 represent course length at 0%, 20%, 40%, 60%, 80%, and 100% of the course module. The merging operation resulted in the formation of the final table called student_info, which consisted of 45 columns, of which 44 were independent, and one feature called final score was dependent.

Whether they are traditional ML multiclass classification algorithms, ensemble multiclass classification algorithms,

TABLE 2. Performance scores of six traditional ML algorithms when trained on all 45 features.

Precision	**LogisticRegression**	**SGDClassifier**	**GaussianNB**	**KNearestNeighbor**	**DecisionTreeClassifier**	**LinearSVC**
Distinction	0.013522	0.221473	0.326298	0.281521	0.436544	0.171850
Fail	0.149760	0.272962	0.252544	0.452503	0.418797	0.191031
Pass	0.933838	0.565594	0.818399	0.749672	0.710565	0.355915
Withdrawn	0.837500	0.713396	0.841990	0.623348	0.692182	0.777463
Averaged	0.834450	0.806309	0.722552	0.605935	0.613550	0.750563
Recall	**LogisticRegression**	**SGDClassifier**	**GaussianNB**	**KNearestNeighbor**	**DecisionTreeClassifier**	**LinearSVC**
Distinction	0.399506	0.170071	0.472019	0.311897	0.423352	0.058632
Fail	0.416987	0.487659	0.421564	0.408864	0.404329	0.475432
Pass	0.647697	0.619832	0.723662	0.690237	0.726229	0.650492
Withdrawn	0.700156	0.696924	0.695402	0.733440	0.697638	0.521749
Averaged	0.648728	0.514441	0.657657	0.602522	0.616329	0.432943
F-score	**LogisticRegression**	**SGDClassifier**	**GaussianNB**	**KNearestNeighbor**	**DecisionTreeClassifier**	**LinearSVC**
Distinction	0.026070	0.075876	0.384997	0.295688	0.429463	0.081371
Fail	0.220034	0.239087	0.315690	0.429422	0.411345	0.202777
Pass	0.764848	0.486444	0.768051	0.718683	0.718255	0.387223
Withdrawn	0.762657	0.674241	0.761682	0.673848	0.694865	0.548618
Averaged	0.719691	0.575717	0.682546	0.602110	0.614775	0.496987
Accuracy	**LogisticRegression**	**SGDClassifier**	**GaussianNB**	**KNearestNeighbor**	**DecisionTreeClassifier**	**LinearSVC**
Distinction	0.394231	0.178669	0.470897	0.311607	0.423484	0.154537
Fail	0.418319	0.325177	0.421875	0.408631	0.404325	0.368436
Pass	0.647719	0.595968	0.723689	0.690256	0.726251	0.616932
Withdrawn	0.700231	0.638215	0.695429	0.733573	0.697767	0.425159
Averaged	0.648728	0.514436	0.657657	0.602522	0.616329	0.432946

TABLE 3. Performance of traditional ML models after feature merging (Pass and Distinction to Pass, Withdrawn and Fail to Fail).

Precision	**LogisticRegression**	**SGDClassifier**	**GaussianNB**	**KNearestNeighbor**	**DecisionTreeClassifier**	**LinearSVC**
Fail	0.885594	0.851161	0.882454	0.897391	0.891028	0.879360
Pass	0.927540	0.891407	0.921391	0.874134	0.870709	0.798653
Averaged	0.906504	0.891603	0.901798	0.886545	0.881548	0.900551
Recall	**LogisticRegression**	**SGDClassifier**	**GaussianNB**	**KNearestNeighbor**	**DecisionTreeClassifier**	**LinearSVC**
Fail	0.931751	0.915074	0.926158	0.888521	0.885128	0.872229
Pass	0.878735	0.851063	0.875084	0.883903	0.877196	0.868030
Averaged	0.905348	0.869696	0.900807	0.886387	0.881416	0.841041
F-score	**LogisticRegression**	**SGDClassifier**	**GaussianNB**	**KNearestNeighbor**	**DecisionTreeClassifier**	**LinearSVC**
Fail	0.908058	0.874950	0.903767	0.892915	0.888041	0.862554
Pass	0.902443	0.859432	0.897624	0.878968	0.873907	0.787617
Averaged	0.905271	0.871779	0.900727	0.886420	0.881441	0.855064
Accuracy	**LogisticRegression**	**SGDClassifier**	**GaussianNB**	**KNearestNeighbor**	**DecisionTreeClassifier**	**LinearSVC**
Fail	0.931764	0.897309	0.926197	0.888544	0.885124	0.829886
Pass	0.878741	0.842034	0.875108	0.883923	0.877210	0.855194
Averaged	0.905348	0.869696	0.900807	0.886387	0.881416	0.841039

or neural networks, all types of ML/DL algorithms require the features to be encoded appropriately into numerical forms for better model training and deployments. The label encoder technique converted all the features with categorical data into a numerical form. The dependent feature called final_result was having four classes, i.e., Pass, Withdrawn, Fail, and Distinction. The final_result was also encoded, and numerical representations were assigned to each class (**'Pass': 2, 'Withdrawn': 3, 'Fail': 1, 'Distinction': 0**). After all the independent and dependent features were encoded correctly, we used six conventional ML algorithms for modeling students' online study behavior and for predicting their performance at different stages of the course. Six traditional multiclass classification algorithms included logistic regression, SGD classifier, Gaussian Naïve Bayes (GNB), K-Nearest Neighbor (KNN), DT classifier, and Linear SVC. All 6 ML models were evaluated in terms of precision, recall, f-score, and accuracy. The score for distinction, fail, pass, and withdrawn classes were also averaged to determine the models' overall performance. Table 2 shows the performance score of all six models when trained on all 45 features. We noticed that the logistic regression classifier showed the best results regarding precision and f-score, whereas GNB showed the best results regarding recall and accuracy. Overall, the pass class had the best results in terms of precision, recall, f-score, and accuracy.

We noticed that the performance results of all predictive models for the Fail class were low. Students belonging to the Fail class are our foremost concern in this study as they are at risk of dropping out and need timely intervention and guidance. To further increase the predictive performance of all six models, we merged the Distinction class with the Pass class and the Fail class with the Withdrawn class, as these classes are almost similar. In table 3, we can observe a decent increase in the performance of all six predictive models. The precision, recall, f-score, and accuracy scores for all six models were greater than 84%, with the logistic regression model showing the best results and linear SVC

TABLE 4. Performance of the logistic regression model when trained on demographic data, 20%, 40%, 60%, 80% and 100% course data.

Precision	Demographic data	20% Course	40% Course	60% Course	80% Course	100% Course
Fail	0.671438	0.720716	0.778379	0.833089	0.872076	0.887199
Pass	0.533930	0.808116	0.851772	0.897184	0.912421	0.926526
Averaged	0.613067	0.767006	0.816563	0.865915	0.892205	0.906758
Recall	**Demographic data**	**20% Course**	**40% Course**	**60% Course**	**80% Course**	**100% Course**
Fail	0.617029	0.807706	0.854465	0.900535	0.917579	0.930990
Pass	0.592373	0.721190	0.774641	0.827772	0.864392	0.880159
Averaged	0.606510	0.761942	0.813027	0.863314	0.891081	0.905747
F-score	**Demographic data**	**20% Course**	**40% Course**	**60% Course**	**80% Course**	**100% Course**
Fail	0.643018	0.761686	0.814608	0.865467	0.894216	0.908550
Pass	0.561546	0.762129	0.811336	0.861043	0.887717	0.902727
Averaged	0.608408	0.761951	0.812952	0.863224	0.890993	0.905672
Accuracy	**Demographic data**	**20% Course**	**40% Course**	**60% Course**	**80% Course**	**100% Course**
Fail	0.617036	0.807685	0.854491	0.900559	0.917574	0.931028
Pass	0.592299	0.721197	0.774605	0.827756	0.864401	0.880148
Averaged	0.606511	0.761943	0.813027	0.863314	0.891081	0.905747

delivering the lowest performance. Based on best performance results, the logistic regression predictive model was used to predict students' performance at different course lengths.

Table 4 shows the performance of the logistic regression model when trained on only demographic data, 20%, 40%, 60%, 80%, and 100% course data. The course at various lengths contains data about assessment scores and clickstream data in the form of students' interactions with the VLE. When trained only on demographic data, the performance score for the logistic regression model was: averaged precision = 0.613067, averaged recall = 0.606510, averaged f-score = 0.608408, and averaged accuracy = 0.606511. When trained on 20% of course length, the results were averaged precision = 0.767006, averaged recall = 0.761942, averaged f-score = 0.761951, and averaged accuracy = 0.761943. Training the logistics regression model only on 20% of course length data gave satisfactory and reasonable results, which indicated that the earliest possible prediction of students' performance is possible even when only 20% of course data is available. Similarly, when trained only on demographic data, the logistics regression model gave more than a 60% performance result score, which indicated that, to some extent, only demographic data could also be used to predict students' performance in the future. As we provided more course data to the logistics regression model, its performance improved, and overall we observed that the averaged prediction accuracy improved from 0.606511 to 0.905747.

B. PHASE 2, BLACK BOX APPROACH: USING ENSEMBLE ML ALGORITHMS FOR PREDICTING STUDENTS' PERFORMANCE AT VARIOUS STAGES OF COURSE LENGTH

Six ensemble ML multiclass classification models selected for predicting students' performance at various percentages of course length included Bagging Classifier, RF, Extra Tree Classifier, Gradient Boosting, AdaBoost Classifier, and Voting Classifier. Like traditional ML models, the six ensemble models were evaluated using precision, recall, f-score, and accuracy metrics. Table 5 shows the performance scores of six ensemble models when trained on all 45 features. Similar to traditional ML models, initially, the students were classified into four classes, i.e., Distinction, Fail, Pass, and Withdrawn. Table 5 shows that overall the gradient boosting showed superior performance compared to the other ensemble models, whereas the AdaBoost classifier showed inferior performance.

To further improve the performance results, a feature engineering process was carried out where Distinction-Pass classes were combined into the Pass class, and Fail-Withdrawn classes were merged into the Fail class. Table 6 displays the results of six ensemble multiclass classification models after performing the feature merging process. We noticed that the performance of all six models improved significantly. Interestingly all six ensemble models showed similar performance results when considering precision, recall, f-score, and accuracy metrics.

For brevity, we selected the RF model to further predict students' performance at various stages of course length. Table 7 illustrates the performance score of the RF model when trained only on demographic data, 20%, 40%, 60%, 80%, and 100% course data. Overall, the average accuracy score improved from 0.594146 to 0.919615. We noted that the RF model's performance results are very similar to the traditional logistic regression model.

C. PHASE 3. EARLIEST POSSIBLE INTERPRETATION OF STUDENTS' STUDY BEHAVIOR USING EXPLAINABLE AI

The primary objective of XAI systems is to make the decisions taken by ML models transparent and understandable to AI experts and non-AI experts to become trustworthy and reliable. That is, an ordinary person should know how and why an AI system makes a particular decision. We selected the RF ensemble classifier to build XAI models to show the effectiveness of various XAI methods and tools in assisting instructors in understanding model prediction results. Different XAI models were created using only demographic data, clickstreams + demographic data, and assessments + clickstreams + demographic data. An XAI model was also created after combining the final four performance classes,

TABLE 5. Students' performance classification into Distinction, Fail, Pass, and Withdrawn categories using ensemble models by using all 45 features.

Precision	BaggingClassifier	RandomForest	ExtraTreeClassifier	GradientBoosting	AdaBoostClassifier	VotingClassifier
Distinction	0.498434	0.449708	0.485215	0.509112	0.451298	0.030130
Fail	0.389022	0.366775	0.383562	0.369181	0.351721	0.226109
Pass	0.881080	0.903020	0.894277	0.896714	0.843484	0.959436
Withdrawn	0.807744	0.830788	0.817817	0.840972	0.717170	0.853178
Averaged	0.753523	0.775948	0.765940	0.778854	0.701279	0.848886
Recall	**BaggingClassifier**	**RandomForest**	**ExtraTreeClassifier**	**GradientBoosting**	**AdaBoostClassifier**	**VotingClassifier**
Distinction	0.624846	0.655595	0.654716	0.666897	0.535540	0.483140
Fail	0.554160	0.580294	0.569537	0.593800	0.457549	0.546345
Pass	0.762992	0.756147	0.761434	0.765135	0.745608	0.684307
Withdrawn	0.748883	0.746810	0.749455	0.748316	0.698050	0.712705
Averaged	0.716258	0.722364	0.721996	0.729267	0.662074	0.681404
F-score	**BaggingClassifier**	**RandomForest**	**ExtraTreeClassifier**	**GradientBoosting**	**AdaBoostClassifier**	**VotingClassifier**
Distinction	0.554431	0.532722	0.556844	0.577207	0.489398	0.056508
Fail	0.456762	0.449277	0.458180	0.455183	0.385507	0.319412
Pass	0.817772	0.823015	0.822458	0.825661	0.791497	0.798808
Withdrawn	0.777095	0.786447	0.782036	0.791895	0.701563	0.776572
Averaged	0.729862	0.740824	0.737476	0.746474	0.676053	0.743439
Accuracy	**BaggingClassifier**	**RandomForest**	**ExtraTreeClassifier**	**GradientBoosting**	**AdaBoostClassifier**	**VotingClassifier**
Distinction	0.625207	0.654791	0.654343	0.667532	0.536374	0.489362
Fail	0.553561	0.579915	0.569234	0.593572	0.442506	0.545704
Pass	0.763049	0.756182	0.761452	0.765153	0.745566	0.684344
Withdrawn	0.748973	0.746857	0.749526	0.748379	0.698985	0.712793
Averaged	0.716258	0.722364	0.721996	0.729267	0.662075	0.681404

TABLE 6. Performance result of six ensemble ML models after feature merging process.

Precision	BaggingClassifier	RandomForest	ExtraTreeClassifier	GradientBoosting	AdaBoostClassifier	VotingClassifier
Fail	0.900634	0.900055	0.899094	0.895553	0.894547	0.887466
Pass	0.952498	0.953348	0.953927	0.956230	0.942826	0.949764
Averaged	0.926695	0.926862	0.926726	0.926274	0.918729	0.919114
Recall	**BaggingClassifier**	**RandomForest**	**ExtraTreeClassifier**	**GradientBoosting**	**AdaBoostClassifier**	**VotingClassifier**
Fail	0.954952	0.955685	0.956194	0.958069	0.945839	0.951814
Pass	0.895490	0.895020	0.894142	0.891057	0.888789	0.882937
Averaged	0.925107	0.925199	0.924953	0.924156	0.917283	0.916853
Recall	**BaggingClassifier**	**RandomForest**	**ExtraTreeClassifier**	**GradientBoosting**	**AdaBoostClassifier**	**VotingClassifier**
Fail	0.954952	0.955685	0.956194	0.958069	0.945839	0.951814
Pass	0.895490	0.895020	0.894142	0.891057	0.888789	0.882937
Averaged	0.925107	0.925199	0.924953	0.924156	0.917283	0.916853
Accuracy	**BaggingClassifier**	**RandomForest**	**ExtraTreeClassifier**	**GradientBoosting**	**AdaBoostClassifier**	**VotingClassifier**
Fail	0.954957	0.955695	0.956180	0.958095	0.945865	0.951820
Pass	0.895502	0.895039	0.894169	0.891090	0.888780	0.882954
Averaged	0.925107	0.925199	0.924953	0.924155	0.917283	0.916853

TABLE 7. Performance result of the RF model on demographic data, 20%, 40%, 80%, and 100% course data.

Precision	No Clickstream, Assessment	20% Course	40% Course	60% Course	80% Course	100% Course
Fail	0.649044	0.743902	0.794922	0.851673	0.881372	0.899130
Pass	0.532977	0.836032	0.887926	0.912219	0.930432	0.942568
Averaged	0.598543	0.792753	0.844071	0.882506	0.906009	0.920763
Recall	**No Clickstream, Assessment**	**20% Course**	**40% Course**	**60% Course**	**80% Course**	**100% Course**
Fail	0.608533	0.835358	0.888128	0.915596	0.934011	0.945926
Pass	0.575881	0.744766	0.794693	0.846061	0.875207	0.893048
Averaged	0.594146	0.787378	0.838830	0.880220	0.904519	0.919615
F-score	**No Clickstream, Assessment**	**20% Course**	**40% Course**	**60% Course**	**80% Course**	**100% Course**
Fail	0.628008	0.786954	0.838900	0.882451	0.906914	0.921927
Pass	0.553432	0.787732	0.838685	0.877862	0.901957	0.917129
Averaged	0.595471	0.787394	0.838827	0.880132	0.904444	0.919550
Accuracy	**No Clickstream, Assessment**	**20% Course**	**40% Course**	**60% Course**	**80% Course**	**100% Course**
Fail	0.608423	0.835356	0.888074	0.915594	0.934044	0.945953
Pass	0.575743	0.744803	0.794706	0.846084	0.875206	0.893084
Averaged	0.594146	0.787378	0.838830	0.880220	0.904519	0.919615

i.e., Distinction, Pass, Fail, and Withdrawn, into two classes, i.e., Pass and Fail. Moreover, different XAI models were created at various course lengths (20%, 40%, 60%, 80%, 100%) to assist instructors in knowing how the study behavior of students varies from the start of the semester to the end of the semester.

TABLE 8. RF model evaluation metrics to be generated by the XAI model.

Methods	Explanations
SHapley Additive exPlanations (SHAP)	Provide individual feature importance in making the final result. Based on cooperative game theory.
Prediction probabilities	Model prediction probabilities vs. observed occurrence
Metrics	Accuracy, Precision, Recall, f-score, ROC-AUC-Score, Pr-AUC-Score, Log_loss
Confusion matrices	To determine the performance of the RF model
ROC Curve	The receiver operating characteristic curve shows the RF model's performance at all classification thresholds.
AUC Curve	The area under the curve shows the area underneath the whole ROC curve.
Permutation importance	Increase or decrease in the RF model performance when a single feature value is arbitrarily reordered.

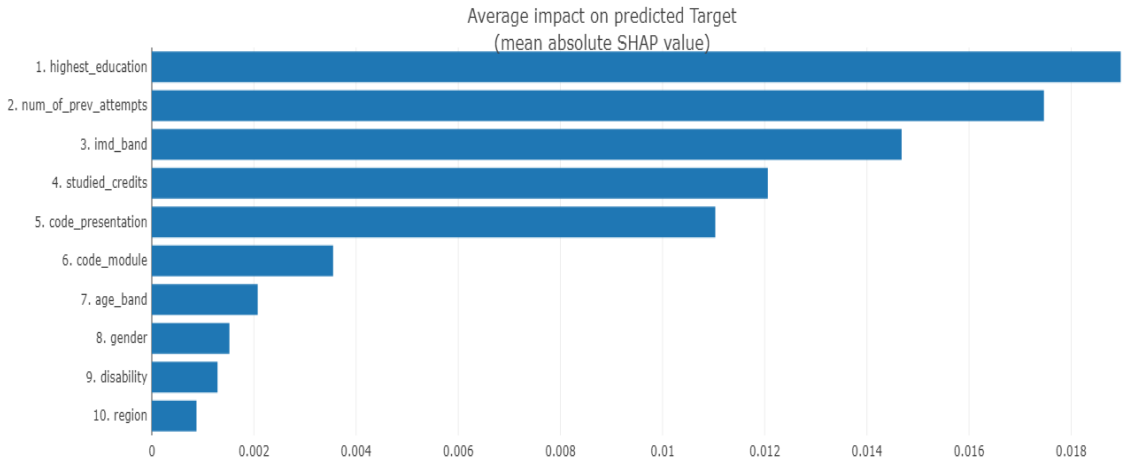


FIGURE 2. Features average SHAP values contribution in predicting Distinction class.

1) CREATING AN XAI MODEL BY UTILIZING ONLY DEMOGRAPHIC FEATURES

We first trained the RF model only on demographic data to determine how the final performance is affected by demographic features. Then the trained RF model is passed to a classifier explainer (an XAI library) to construct the XAI model. The XAI model provided information presented in table 8 for understanding how the prediction was made for Distinction, Pass, Fail and Withdrawn by the RF model when only demographic features were used.

2) DETERMINING FEATURE IMPORTANCE BY MEAN ABSOLUTE SHapley ADDITIVE exPlanations (SHAP) VALUE

SHAP values determine how much an individual feature relatively contributes to predicting a class or what is the impact of a particular feature on the final result. Figure 2 presents each feature's average SHAP contribution in predicting students' performance in the Distinction class. We can observe that when only demographic characteristics are considered, a student's previous highest education impacts their grades most.

When setting the cutoff prediction probability to 0.46 and the cutoff percentile of samples to.9, we obtained a list of XAI model performance metrics for the Pass class shown in table 9.

Figure 3 shows the trade-off between false positives and false negatives in the form of the ROC-AUC curve. Similarly, the trade-off between precision and recall is presented in figure 4 when predicting the Pass class.

TABLE 9. XAI model performance metrics for the Pass class when considering all independent features.

metric	Score
accuracy	0.619
precision	0.542
recall	0.09
f1	0.154
roc_auc_score	0.608
pr_auc_score	0.478
log_loss	0.651

In addition, an interaction-dependent plot was generated by the XAI model as shown in figure 5. The interaction dependence plots show the relation between features and Shap interaction values. Figure 5 shows how the number_of_previous_attempts feature interacted with highest_education, keeping number_of_previous_attempts independent. The values above 0 indicated that the features positively impact predicting Pass grade (*Pass grade is selected as an example*). In contrast, the values below 0 showed that the features negatively impacted predicting the Pass grade, which implies that these negative values were used to predict other grades. For conciseness, only these two features are demonstrated. Similar plots can also be generated for other demographic features.

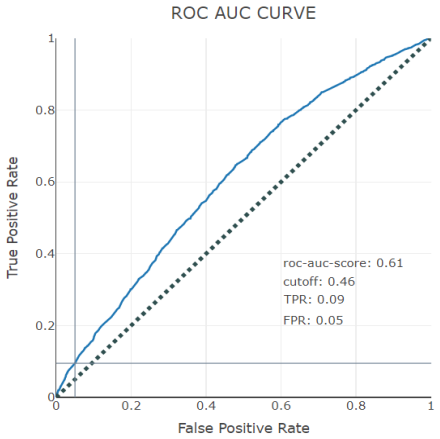


FIGURE 3. Trade-off between false positives and false negatives in the form of the ROC-AUC curve when predicting Pass class.

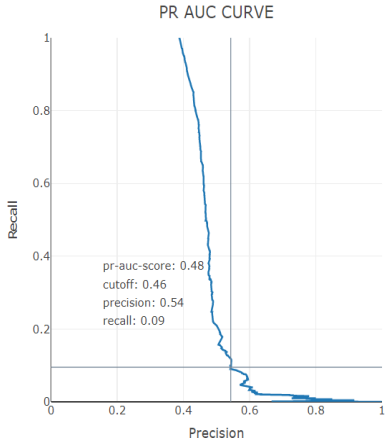


FIGURE 4. Trade-off between precision and recall for the RF model when predicting the Pass class.

D. PHASE 4. CREATING AN XAI MODEL BY UTILIZING DEMOGRAPHIC AND CLICKSTREAM FEATURES

To know how much clickstream data impacted students' performance, we added clickstream features (sum clicks and mean clicks) to the demographic data. Once again, the RF model was built by keeping the training set size to 80% and the testing set size to 20%. For generating the XAI model, the RF model was passed to the explainer classifier (Python library) for feature interpretation and contribution to predicting the final scores.

The figures 6a and 6b show the features for Distinction and Pass classes. In contrast, figures 6c and 6d show the features for Fail and Withdrawn classes, sorted from most important to least important by mean absolute shap values for the final four classes.

We can observe that the top three critical features for predicting the Distinction class are sum_clicks, highest education, and mean_clicks. For the Pass category, the top three essential features are sum_clicks, mean_clicks, and code_module. For the Fail class, sum_clicks, mean_clicks, and highest education had a significant effect. Lastly, the top three critical features for the Withdrawn class include sum_clicks, mean_clicks, and studied credit hours. It can be concluded that clickstream data in the form of sum_clicks and mean_clicks features significantly impact the students' final performance.

TABLE 10. Features contributions to predicting the Distinction and Pass class on a random observation.

	Distinction Class	Pass Class
Reason	**Effect**	**Effect**
Average of population	9.28%	37.74%
sum_clicks100 = 1294.0	+5.35%	+14.25%
mean_clicks100 = 3.24526448128143	+1.16%	+3.19%
highest_education = 0.0	+1.4%	+0.77%
studied_credits = 60.0	-0.06%	+1.08%
code_presentation = 3.0	+0.26%	-0.39%
num_of_prev_attempts = 0.0	+0.24%	+0.23%
code_module = 1.0	+0.09%	+1.71%
imd_band = 8.0	+0.62%	+0.29%
gender = 1.0	-0.11%	-0.33%
disability = 0.0	+0.05%	+0.15%
age_band = 0.0	-0.03%	+0.02%
region = 8.0	+0.0%	+0.0%
Other features combined	+0.0%	+0.0%
Final prediction	18.24%	58.71%

TABLE 11. Features contributions to predicting the Fail and Withdrawn class on a random observation.

	Fail Class	Withdrawn Class
Reason	Effect	Effect
Average of population	21.5%	31.48%
sum_clicks100 = 1294.0	-6.32%	-13.28%
mean_clicks100 = 3.24526448128143	-0.2%	-4.14%
highest_education = 0.0	-1.38%	-0.79%
studied_credits = 60.0	+0.31%	-1.34%
code_presentation = 3.0	-0.58%	+0.7%
num_of_prev_attempts = 0.0	-0.42%	-0.05%
code_module = 1.0	-0.74%	-1.06%
imd_band = 8.0	-0.54%	-0.37%
gender = 1.0	+0.08%	+0.36%
disability = 0.0	-0.04%	-0.16%
age_band = 0.0	+0.02%	-0.01%
region = 8.0	+0.0%	-0.01%
Other features combined	+0.0%	+0.0%
Final prediction	11.7%	11.35%

Tables 10 and 11 show each feature's contribution to the prediction of a particular observation when considering the Distinction, Pass, Fail, and Withdrawn classes. These findings can help both AI and non-AI experts in describing precisely how each prediction has been made from all the distinctive features in the model. Positive shap values for the four classes positively impact the final predictions, which will lead the model to predict the final performance as Distinction and Pass. The negative shap values for the four classes have a negative impact on the final prediction, which will lead the model to predict the final performance as Fail or Withdrawn.

E. PHASE 5. CREATING AN XAI MODEL BY ADDING ASSESSMENT SCORE

Figures 7a, 7b, 7c, and 7d show features sorted from most important to least important in predicting the Distinction, Pass, Fail, and Withdrawn classes. We noticed that features

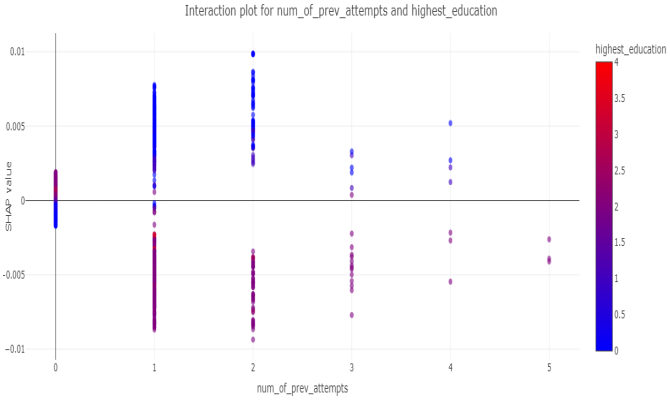
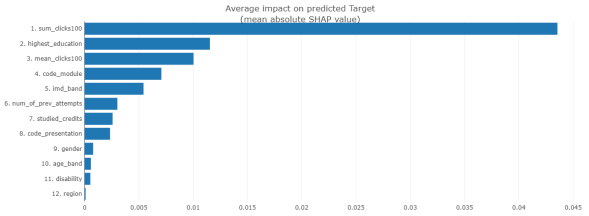
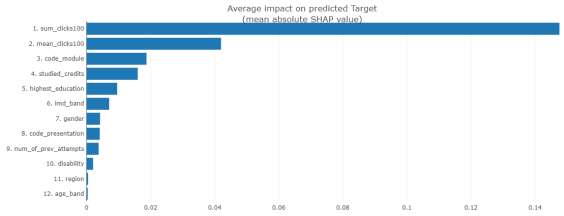


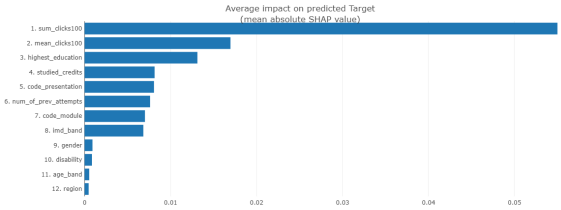
FIGURE 5. Interaction dependence plots generated the XAI model showing the relationship between features and Shap interaction values.



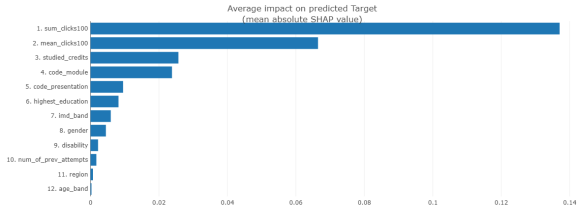
(a) Feature importance/weights sorted from most important to least important for predicting Distinction class



(b) Feature importance/weights sorted from most important to least important for predicting Pass class



(c) Feature importance/weights sorted from most important to least important for predicting Fail class



(d) Feature importance/weights sorted from most important to least important for predicting Withdrawn class

FIGURE 6. Features sorted from most important to least important by mean absolute shap values.

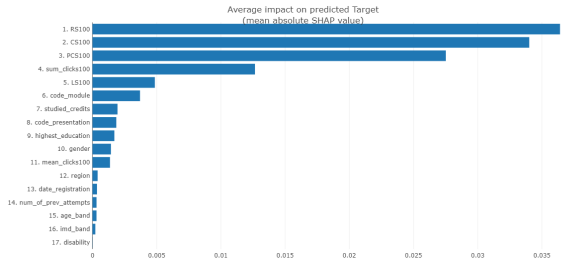
other than demographic features such as RS100, CS100, PCS100, sum_clicks100, LS100, and studied_credits significantly impact the final grade when considering all four classes. This concludes that students' performance improves by adding assessment features to the XAI model.

Table 12 shows each feature's contribution to predicting the Distinction and Pass classes when an observation is selected randomly. Similarly, table 13 shows each feature's contribution to predicting the Fail and Withdrawn class when an observation is chosen randomly. From the results, we concluded that assessments score has the highest impact in predicting students' final performance.

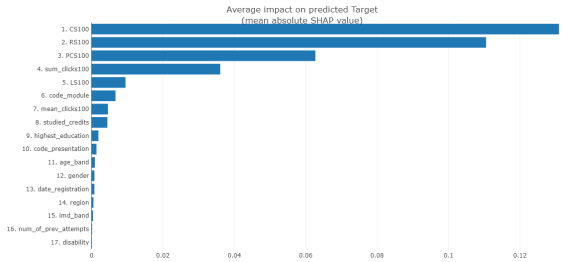
1) CREATING AND INTERPRETING XAI MODELS AT DIFFERENT PERCENTAGES OF COURSE LENGTH

Various XAI models were created at different percentages of course length to interpret in a human-readable way which features influence students' study behavior most. Once again, to improve the accuracy of the RF model, the Pass class was merged into the Distinction class, whereas the Withdrawn class was combined into the Fail class. The Pass class was encoded with 0, and the Fail class was given 1. The goal of creating XAI models at various course lengths is to determine the overall performance of models and to investigate how prediction is made for individual observation. Confusion matrices determine the overall performance of different XAI models **(global explainability)**, and the prediction for each observation is determined by the weight or importance of each feature **(local explainability)**.

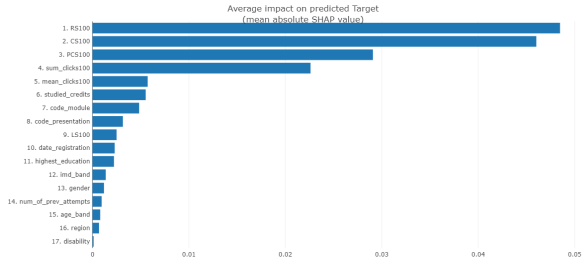
Table 14 displays the RF model metrics scores extracted by the XAI model when trained on 20%, 40%, 60%, 80% and 100% course data. We can observe that adding more course data increases the scores for accuracy, precision, recall, f1, roc_auc_score, and pr_auc_score, whereas the log_loss value decreases. The results imply that when provided more course data, the RF model train and generalizes well, thus becoming more reliable. ROC_AUC_Score is the Area Under the Curve (AUC) of the Receiver Characteristics Operator (ROC).



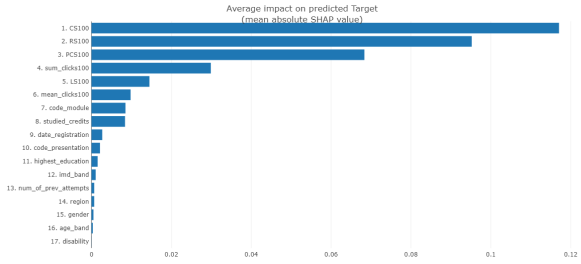
(a) Feature importance/weights sorted from most important to least important for predicting Distinction class when assessment scores are also considered



(b) Feature importance/weights sorted from most important to least important for predicting Pass class when assessment scores are also considered



(c) Feature importance/weights sorted from most important to least important for predicting Fail class when assessment scores are also considered



(d) Feature importance/weights sorted from most important to least important for predicting Withdrawn class when assessment scores are also considered

FIGURE 7. Features sorted from most important to least important by shap values when assessment scores features are added to demographics and clickstream features.

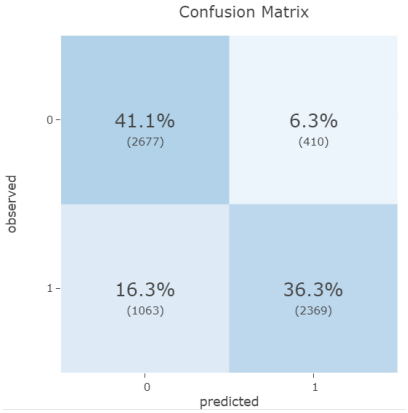
TABLE 12. Features' contribution to predicting the Distinction and Pass classes on a randomly selected observation.

Distinction Class		Pass Class	
Reason	**Effect**	**Reason**	**Effect**
Average of population	9.34%	Average of population	9.34%
RS100 = 90.5	+15.38%	RS100 = 94.0	+16.63%
CS100 = 91.5	+14.85%	CS100 = 93.76	+15.79%
PCS100 = 91.5	+10.13%	PCS100 = 93.76	+12.28%
sum_clicks100 = 6746.0	+3.41%	sum_clicks100 = 766.0	+0.96%
mean_clicks100 = 4.661417806	+0.61%	LS100 = 0.0	-0.51%
code_presentation = 2.0	+0.57%	code_presentation = 3.0	+0.42%
imd_band = 1.0	-0.28%	age_band = 0.0	-0.27%
LS100 = 0.0	-0.28%	studied_credits = 60.0	-0.21%
studied_credits = 60.0	-0.26%	mean_clicks100 = 2.7404776	+0.21%
num_of_prev_attempts = 0.0	+0.24%	num_of_prev_attempts = 0.0	+0.17%
highest_education = 0.0	+0.22%	highest_education = 0.0	+0.07%
code_module = 5.0	+0.15%	date_registration = -31.0	+0.04%
age_band = 0.0	-0.06%	region = 10.0	-0.02%
date_registration = -99.0	-0.05%	imd_band = 6.0	+0.02%
region = 5.0	+0.02%	code_module = 4.0	-0.01%
gender = 1.0	+0.0%	gender = 1.0	+0.0%
disability = 0.0	-0.0%	disability = 0.0	-0.0%
Other features combined	+0.0%	Other features combined	+0.0%
Final prediction	53.99%	Final prediction	54.91%

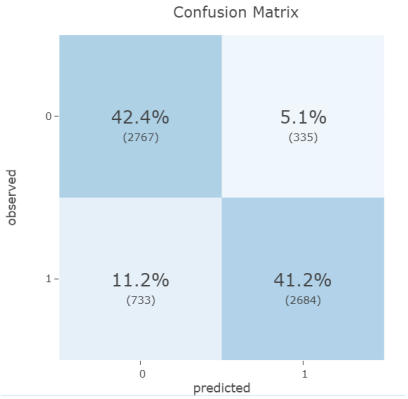
A higher roc_auc score helps us visualize how well the RF model is performing. PR_AUC_Score is the precision-recall area under the curve. Similar to roc_auc_score, the higher the pr_auc_score, the better the RF model performs for accurately predicting the Pass class. We also observe that the log_loss value decreases for the RF model when trained on more data which implies that the difference between the observed and predicted value is minimized, thus increasing the RF model's accuracy.

F. PHASE 6. GENERATING CONFUSION MATRICES FOR GLOBAL EXPLAINABILITY

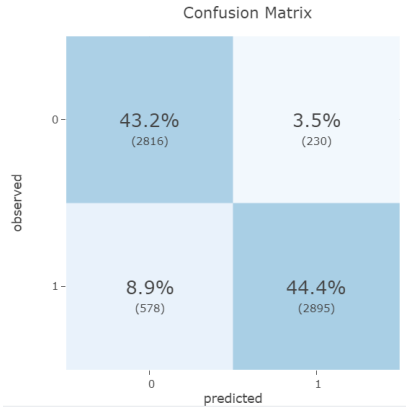
Figure 8 shows confusion matrices showing the number of true positives, true negatives, false negatives, and false positives generated by the XAI model for the RF model when trained only on 20%, 40%, 60%, 80% and 100% course data. The number of false negatives and false positives adversely affects the RF model, especially in the deployment phase. Therefore, the number of false negatives and false



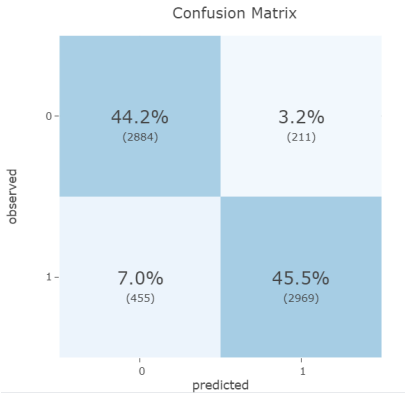
(a) Confusion matrix generated for the RF model when 20% course data is considered



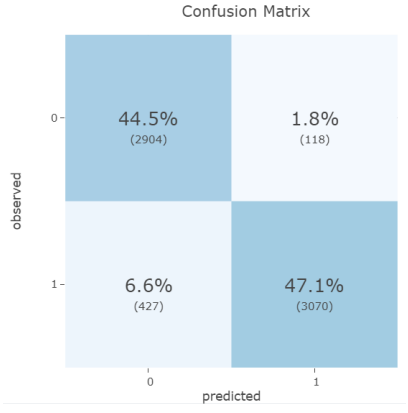
(b) Confusion matrix generated for the RF model when 40% course data is considered



(c) Confusion matrix generated for the RF model when 60% course data is considered



(d) Confusion matrix generated for the RF model when 80% course data is considered



(e) Confusion matrix generated for the RF model when 100% course data is considered

FIGURE 8. Features sorted from most important to least important by shap values when assessment scores features are added to demographics and clickstream features.

positives should be kept low. We observed an increase in the number of true positives coupled with true negatives and a decrease in the number of false positives along with false negatives for the RF model when trained on 40%, 60%, 80%, and 100% of course data. Confusion matrices and feature importance values can help AI experts explain to non-AI experts what features are important for decision-making, to provide global explainability and how the complexity of the model can be reduced while keeping the needed accuracy.

TABLE 13. Features' contribution to predicting the Fail and Withdrawn classes on a randomly selected observation.

Fail Class		Withdrawn Class	
Reason	**Effect**	**Reason**	**Effect**
Average of population	21.62%	Average of population	31.14%
RS100 = 68.08333333333333	-3.67%	RS100 = 68.08333333333333	-7.83%
CS100 = 71.5	-4.76%	CS100 = 71.5	-10.55%
PCS100 = 71.5	-1.13%	PCS100 = 71.5	-6.69%
sum_clicks100 = 1759.0	-1.54%	sum_clicks100 = 1759.0	-2.36%
mean_clicks100 = 1.94092457	-0.06%	mean_clicks100 = 1.940924572	-0.44%
studied_credits = 120.0	-0.16%	studied_credits = 120.0	+0.65%
code_module = 5.0	+0.75%	code_module = 5.0	+0.36%
code_presentation = 3.0	-0.19%	code_presentation = 3.0	+0.12%
LS100 = 4.0	+0.16%	LS100 = 4.0	-2.36%
date_registration = -9.0	+0.14%	date_registration = -9.0	-0.18%
highest_education = 0.0	-0.11%	highest_education = 0.0	-0.07%
imd_band = 4.0	-0.03%	imd_band = 4.0	-0.0%
gender = 1.0	+0.3%	gender = 1.0	+0.02%
num_of_prev_attempts = 0.0	-0.03%	num_of_prev_attempts = 0.0	+0.0%
age_band = 0.0	+0.03%	age_band = 0.0	-0.01%
region = 2.0	-0.04%	region = 2.0	+0.04%
disability = 0.0	+0.0%	disability = 0.0	-0.0%
Other features combined	+0.0%	Other features combined	+0.0%
Final prediction	11.29%	Final prediction	1.83%

TABLE 14. RF model metrics scores extracted by the XAI model when trained on 20%, 40%, 60%, 80% and 100% course data for predicting the Pass class.

Course length	20%	40%	60%	80%	100%
metric	Score	Score	Score	Score	Score
accuracy	0.774	0.836	0.876	0.898	0.916
precision	0.716	0.791	0.83	0.864	0.872
recall	0.867	0.892	0.924	0.932	0.961
f1	0.784	0.838	0.875	0.896	0.914
roc_auc_score	0.855	0.91	0.944	0.958	0.967
pr_auc_score	0.803	0.873	0.919	0.94	0.945
log_loss	0.465	0.384	0.32	0.28	0.242

TABLE 15. RF model metrics scores extracted by the XAI model when trained on 20%, 40%, 60%, 80% and 100% course data for predicting the Fail class.

Course length	20%	40%	60%	80%	100%
metric	Score	Score	Score	Score	Score
accuracy	0.774	0.836	0.876	0.898	0.916
precision	0.852	0.889	0.926	0.934	0.963
recall	0.69	0.785	0.834	0.867	0.878
f1	0.763	0.834	0.878	0.899	0.918
roc_auc_score	0.855	0.91	0.944	0.958	0.967
pr_auc_score	0.891	0.934	0.958	0.969	0.978
log_loss	0.465	0.384	0.32	0.28	0.242

The number of false negatives and false positives adversely affects the RF model, especially in the deployment phase. Therefore, the number of false negatives and false positives should be kept low. We will observe whether the false negatives and positives increase or decrease when more course data is used for the RF model training.

Table 15 shows the metrics score of the RF model predicting the Fail class when trained on 20%, 40%, 60%, 80%, and 100% course data. Similar to the performance of the RF model for the Pass class, we noticed that the performance of the RF model increases for the Fail class for various metrics when it is trained on more and more course data. We observed that the score for accuracy, precision, recall, f1, roc_auc, and pr_auc has increased upon training the RF model on more course data. It is noticeable that even at 20% and 40% of the course data, the RF model shows acceptable performance and can be used by the instructors to intervene with the students as early as possible in the course for needed guidance and feedback. The log_loss values gradually decrease for the RF model when trained on more course data, indicating that the model becomes more mature and reliable at the end of the course. We also observe that the difference between the performance scores of the RF model for the Pass and Fail class is negligible, indicating that the model performance for predicting both classes is almost the same.

G. PHASE 7. LOCAL EXPLAINABILITY AT DIFFERENT PERCENTAGES OF THE COURSE LENGTH

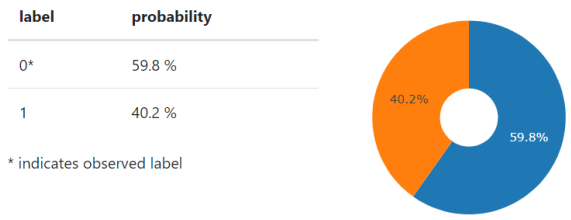
In the last stage of this research study, we tried to explain the decision-making process of the RF model by considering a single observation for the Pass and Fail class at 20%, 40%, 60%, 80%, and 100% of course data. The XAI model will be understandable and transparent to instructors and students as the model explains the prediction of a single observation. With local explainability, instructors can measure how a single feature of the dataset influences the final output and why a particular student was classified into the Pass or Fail class. We selected five random observations for 20%, 40%, 60%, 80%, and 100% course length for the Pass class to observe features' weights and importance in predicting the Pass class.

1) LOCAL EXPLAINABILITY OF THE PASS CLASS AFTER 20%, 40%, 60%, 80% AND 100% COURSE COMPLETION

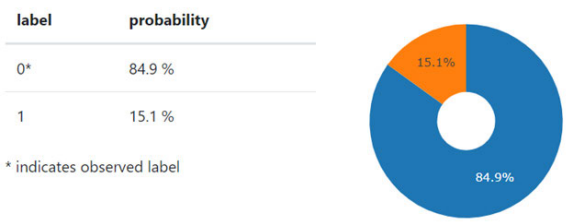
Table 16 shows the feature weights of the RF model for predicting the **Pass class** when a single observation is

TABLE 16. RF model performance for predicting the Pass class on randomly selected observations when trained on 20%, 40%, 60%, 80% and 100% course data.

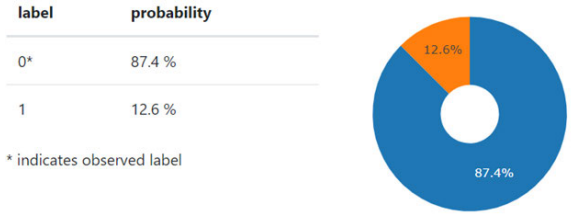
Local explainability for the Pass class: RF model performance on randomly selected observations									
RF model performance on 20% course data		RF model performance on 40% course data		RF model performance on 60% course data		RF model performance on 80% course data		RF model performance on 100% course data	
Reason	**Effect**	**Reason**	**Effect**	**Reason**	**Effect**	**Reason**	**Effect**	**Reason**	**Effect**
Average of population	46.96%	Average of population	47.29%	Average of population	47.14%	Average of population	47.24%	Average of population	47.19%
RS20 = 67.5	+4.99%	RS40 = 85.33	+16.91%	RS60 = 89.42	+24.48%	RS80 = 82.6	+27.02%	RS100 = 91.11	+27.35%
sum_clicks20 = 257.0	+3.29%	sum_clicks40 = 4837.0	+11.83%	sum_clicks60 = 2719.0	+7.46%	sum_clicks80 = 2845.0	+11.42%	sum_clicks100 = 1670.0	+9.22%
sum_clicks0 = 92.0	+1.88%	sum_clicks0 = 496.0	+3.31%	LS60 = 3.0	+4.26%	mean_clicks80 = 3.472	+2.14%	mean_clicks100 = 2.94	+2.28%
highest_education = 0.0	+1.77%	mean_clicks0 = 4.437	+1.52%	sum_clicks0 = 159.0	+2.01%	mean_clicks0 = 2.386	+1.48%	highest_education = 0.0	+1.27%
code_module = 3.0	-1.21%	imd_band = 10.0	+1.38%	mean_clicks0 = 6.160	+1.78%	sum_clicks0 = 214.0	+1.17%	code_module = 6.0	+1.05%
LS20 = 0.0	+0.79%	mean_clicks40 = 4.241	+0.99%	highest_education = 2.0	-1.66%	highest_education = 2.0	-0.53%	sum_clicks0 = 73.0	+1.01%
gender = 1.0	-0.75%	highest_education = 0.0	+0.94%	mean_clicks60 = 5.268	+1.29%	code_module = 5.0	-0.49%	mean_clicks0 = 2.776	+0.7%
num_of_prev_attempts = 0.0	+0.61%	LS40 = 0.0	+0.73%	code_presentation = 0.0	-0.67%	imd_band = 10.0	+0.43%	studied_credits = 30.0	+0.65%
mean_clicks20 = 1.928	+0.61%	num_of_prev_attempts = 0.0	+0.23%	imd_band = 5.0	+0.54%	num_of_prev_attempts = 0.0	+0.37%	imd_band = 6.0	+0.56%
imd_band = 5.0	+0.5%	studied_credits = 60.0	+0.21%	code_module = 1.0	+0.41%	LS80 = 0.0	-0.27%	LS100 = 0.0	+0.33%
mean_clicks0 = 1.81712	+0.48%	code_module = 5.0	-0.2%	num_of_prev_attempts = 0.0	+0.19%	studied_credits = 60.0	+0.22%	region = 9.0	+0.11%
studied_credits = 120.0	-0.4%	code_presentation = 0.0	-0.15%	gender = 0.0	+0.09%	gender = 1.0	-0.16%	disability = 0.0	+0.11%
code_presentation = 3.0	+0.27%	gender = 1.0	-0.07%	studied_credits = 60.0	-0.03%	code_presentation = 1.0	+0.13%	num_of_prev_attempts = 0.0	+0.09%
disability = 0.0	+0.18%	region = 4.0	-0.05%	age_band = 1.0	+0.03%	date_registration = -24.0	+0.08%	gender = 0.0	+0.09%
age_band = 0.0	-0.11%	date_registration = -289.0	-0.04%	date_registration = -22.0	+0.02%	region = 2.0	-0.02%	code_presentation = 2.0	+0.09%
region = 3.0	-0.04%	disability = 0.0	+0.02%	disability = 0.0	+0.02%	disability = 0.0	+0.01%	date_registration = -54.0	+0.02%
date_registration = -30.0	-0.02%	age_band = 1.0	+0.0%	region = 7.0	+0.01%	age_band = 1.0	+0.0%	age_band = 1.0	-0.01%
Other features combined	+0.0%	Other features combined	+0.0%	Other features combined	+0.0%	Other features combined	+0.0%	Other features combined	+0.0%
Final prediction	59.82%	Final prediction	84.86%	Final prediction	87.37%	Final prediction	90.22%	Final prediction	92.1%



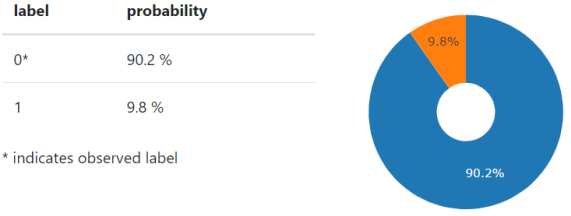
(a) Prediction probability of the Pass class at 20% course length



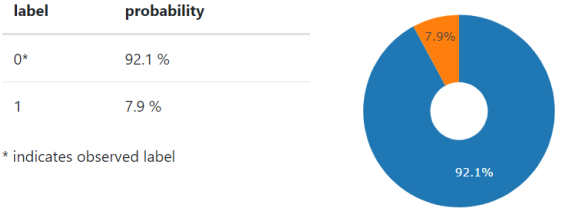
(b) Prediction probability of the Pass class at 40% course length



(c) Prediction probability of the Pass class at 60% course length



(d) Prediction probability of the Pass class at 80% course length



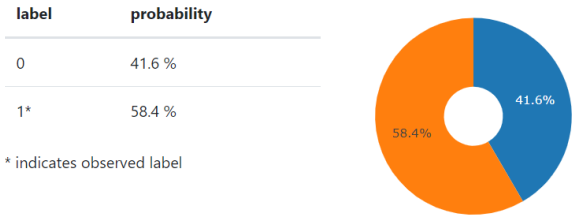
(e) Prediction probability of the Pass class at 100% course length

FIGURE 9. Prediction probabilities of the RF model at 20%, 40%, 60%, 80% and 100% course length on a single randomly selected observation. Pass class is encoded as 0 and Fail as 1.

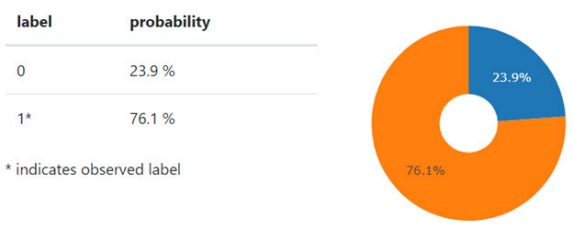
selected, whereas figure 9 shows the prediction probability of each observed target class at 20%, 40%, 60%, 80% and 100% course length. The single observation prediction results revealed that even at 20% of course completion, the top three

TABLE 17. RF model performance for predicting the Fail class on randomly selected observations when trained on 20%, 40%, 60%, 80% and 100% course data.

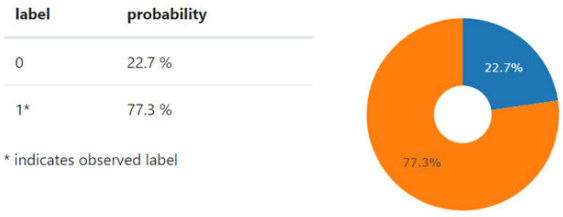
Local explainability for the Fail class: RF model performance on randomly selected observations									
RF model performance on 20% course data		RF model performance on 40% course data		RF model performance on 60% course data		RF model performance on 80% course data		RF model performance on 100% course data	
Reason	**Effect**	**Reason**	**Effect**	**Reason**	**Effect**	**Reason**	**Effect**	**Reason**	**Effect**
Average of population	53.04%	Average of population	52.71%	Average of population	52.86%	Average of population	52.76%	Average of population	52.81%
RS20 = 60.0	+2.77%	RS40 = 0.0	+23.3%	RS60 = 24.25	+27.87%	RS80 = 14.5	+26.96%	RS100 = 13.5	+30.88%
highest_education = 2.0	+2.33%	sum_clicks40 = 205.0	+5.72%	sum_clicks60 = 725.0	-8.62%	sum_clicks0 = 0.0	+2.61%	sum_clicks100 = 519.0	-2.49%
imd_band = 2.0	+1.26%	sum_clicks0 = 87.0	-2.9%	sum_clicks0 = 0.0	+3.14%	mean_clicks80 = 3.40	-1.51%	code_presentation = 0.0	+1.99%
LS20 = 1.0	+1.21%	code_module = 3.0	+1.53%	mean_clicks60 = 3.08	-2.7%	LS80 = 0.0	+1.23%	imd_band = 0.0	+1.95%
code_module = 3.0	+1.19%	highest_education = 0.0	-1.29%	LS60 = 0.0	+2.04%	code_module = 4.0	+1.12%	code_module = 5.0	+1.82%
sum_clicks20 = 242.0	-1.15%	mean_clicks0 = 1.83	-1.01%	mean_clicks0 = 0.0	+1.96%	mean_clicks0 = 0.0	+0.84%	sum_clicks0 = 1.0	+1.66%
gender = 0.0	-1.03%	imd_band = 9.0	-0.91%	code_module = 5.0	+1.18%	imd_band = 5.0	-0.71%	mean_clicks100 = 3.17	-1.57%
num_of_prev_attempts = 0.0	-0.75%	mean_clicks40 = 1.74	-0.8%	highest_education = 0.0	-1.04%	highest_education = 2.0	+0.32%	LS100 = 0.0	+1.51%
mean_clicks0 = 2.61	-0.47%	studied_credits = 60.0	-0.55%	gender = 1.0	+0.6%	sum_clicks80 = 410.0	-0.17%	mean_clicks0 = 1.0	+1.49%
mean_clicks20 = 2.06	-0.46%	LS40 = 0.0	+0.54%	studied_credits = 60.0	+0.21%	num_of_prev_attempts = 0.0	-0.14%	highest_education = 2.0	+0.46%
code_presentation = 2.0	+0.24%	num_of_prev_attempts = 0.0	-0.23%	num_of_prev_attempts = 0.0	-0.15%	studied_credits = 90.0	+0.09%	gender = 1.0	+0.22%
sum_clicks0 = 17.0	+0.16%	region = 7.0	-0.15%	imd_band = 8.0	-0.09%	gender = 1.0	+0.06%	region = 10.0	-0.12%
disability = 0.0	-0.15%	code_presentation = 2.0	+0.06%	code_presentation = 2.0	+0.04%	code_presentation = 1.0	-0.02%	num_of_prev_attempts = 0.0	-0.11%
studied_credits = 60.0	+0.11%	date_registration = -22.0	+0.05%	date_registration = -45.0	+0.03%	region = 0.0	-0.01%	studied_credits = 60.0	-0.1%
region = 3.0	+0.03%	gender = 1.0	+0.04%	region = 6.0	-0.02%	disability = 0.0	-0.01%	disability = 0.0	-0.05%
age_band = 0.0	+0.03%	disability = 0.0	-0.02%	age_band = 0.0	+0.01%	date_registration = -63.0	+0.0%	date_registration = -30.0	+0.01%
date_registration = -107.0	+0.03%	age_band = 0.0	+0.0%	disability = 0.0	-0.01%	age_band = 0.0	-0.0%	age_band = 0.0	+0.0%
Other features combined	+0.0%	Other features combined	+0.0%	Other features combined	+0.0%	Other features combined	+0.0%	Other features combined	+0.0%
Final prediction	58.38%	Final prediction	76.07%	Final prediction	77.3%	Final prediction	83.41%	Final prediction	90.35%



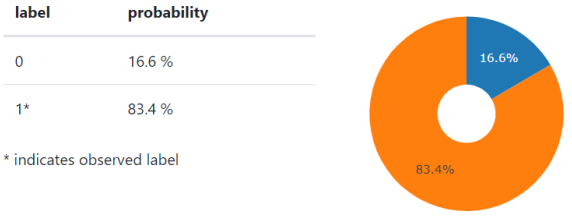
(a) Prediction probability of the Fail class at 20% course length



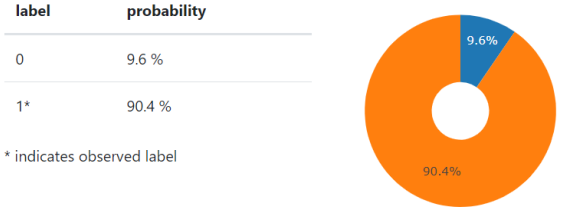
(b) Prediction probability of the Fail class at 40% course length



(c) Prediction probability of the Fail class at 60% course length



(d) Prediction probability of the Fail class at 80% course length



(e) Prediction probability of the Fail class at 100% course length

FIGURE 10. Prediction probabilities of the RF model at 20%, 40%, 60%, 80% and 100% course length on a single randomly selected observation with the Fail class as the observed label. Fail class is encoded as 1 and Pass as 0.

important features impacting the students' performance were assessment score, number of clicks, and previous highest education. Although assessment score, number of clicks, and highest education were the top three important features, their overall effect on students' performance was negligible (RS20 = +4.99%, sum_clicks20 = +3.29%, and

highest_education = +1.77%) as the RF model was training on only 20% of course data. At 20% of course completion, the RF model will predict the Pass class with 59.8% probability and the Fail class with 40.2% probability.

Figures 9a, 9b, 10c, 10d, 10e present the prediction probabilities of the RF model when the Pass class (encoded as 0) is selected as an observed class for 20%, 40%, 60%, 80% and 100% course length. Each observation is selected randomly at 20%, 40%, 60%, 80%, and 100% course length. At 20% course length, the RF model will predict the Pass class with a 59.8% probability. At 40% course length, the RF model will predict the Pass class with 84.9% probability. We noticed that at 40% course length, the prediction probability of the RF model for randomly selected observation is noticeable and considerable. Thus at 40% course length, the instructor can know how the student will perform in the future with 84.9% accuracy. Similarly, at 60%, 80%, and 100% of course length, the RF model prediction probability has increased from 87.4% to 92.1%.

2) LOCAL EXPLAINABILITY OF FAIL CLASS AFTER 20%, 40%, 60%, 80% AND 100% COURSE COMPLETION

Table 17 shows each feature's importance in predicting the Fail class at different percentages of course length. A random observation is selected at each percentage with the Fail class as an observed label. Unlike the results of the Pass class, the important features for predicting the Fail class are different. At 20% course length, the top three important features are assessment scores, highest education, and immigration band. In contrast, the top three important features for predicting the Pass class at 20% course length were assessment score, sum_clicks, and highest education. The results revealed that students classified into the Pass class had more clicks at 20% course length. Referring to the RF model performance at 40% course length, we noticed that other than the average population feature, the top three important features were assessment scores, sum_clicks40 and sum_clicks0. The values for the sum_clicks0 and mean clicks are negative, meaning these features increase the RF model log loss. The features having negative values do not help the RF model in its training process, and the model is not using these features well. At 60%, 80%, and 100% course length, the values for most clickstream features are negative, which means that the students who are classified in the Fail class have a low number of clicks, thus less interaction with the online system.

Referring to figure 10, we noticed that the performance of the RF model for predicting the Fail class is similar to that of predicting the Pass class on random observations at multiple course lengths. The performance accuracy increases with the addition of more and more data at multiple course lengths. At 40% course length, the performance accuracy of the RF is 76.1% for predicting the Fail class (Fail class encoded with 1 is the observed label). The results at 40% course length are encouraging, which means that the earliest possible identification of students at risk of failure is possible, therefore, can be intervened for needed help and guidance to stay on the right track. At 60%, 80%, and 100%, the prediction probability of the Fail class increase from 77.3% to 83.4% and then to 90.4%.

V. CONCLUSION, LIMITATIONS, AND FUTURE WORK

In this study, we proposed the XAI model to facilitate and help instructors in interpreting online students' study behavior. The main objective of this research was to make ML models easy to understand in a human-readable way. Therefore, instructors can know how a particular student was classified into a specific class and how the ML model made various decisions. Initially, six traditional ML models and six ensemble ML models were trained on the OULA dataset to know which model gives the best results in terms of precision, recall, f-score, and accuracy. The ML models' performance results revealed that among traditional ML models, the logistic regression model gave the best results and among ensemble ML models, overall, the RF model showed the best results.

For brevity and due to time constraints, between the logistic regression and the RF model, we selected the RF model as a candidate model for the XAI model to explain how students were classified into various groups and how different decisions were made in a human interpretable way. The purpose of the XAI model was to explain the working of the RF model by using various graphs, charts, and tables that are easy to understand. The XAI model provided results in the form of feature importance, SHAP values, prediction probabilities, metrics such as accuracy, precision, recall, f-score, confusion matrices, ROC-AUC curves, and permutation importance. By utilizing the OULA dataset, initially, the RF model was trained only on demographic features to determine whether, at the start of the semester, students' performance can be predicted with acceptable accuracy. Gradually, clickstream and assessment features were also added to determine how the RF model performance increases after adding more features. The RF model was provided to the XAI model as an input to generate and provide the model explainability and internal working. Various XAI models were also created at 20%, 40%, 60%, 80%, and 100% of course length for the earliest possible interpretation and understanding of students' study behavior.

For understanding the overall performance of the RF model and for global explainability, confusion matrices were created at 20%, 40%, 60%, 80%, and 100% of course length. The purpose of generating confusion matrices was to determine to what extent each feature contributes to the model decision by utilizing all the data. By performing global explainability, the instructors will come to know about the most important features to predict students' performance. For understanding the root cause of a particular decision made by the RF model, we performed local explainability both for the Pass and the Fail class at 20%, 40%, 60%, 80%, and 100% of course length. Local explainability will help instructors to get to the

bottom of which feature was most impactful in categorizing a particular student into the Pass or the Fail class.

Due to time constraints, we were not able to leverage the power of deep neural networks such as ANNs, LSTM, and transformers in modeling the study behavior of online students and predicting their performance. Moreover, we also did not perform experiments regarding which deep neural network is accurate as well as interpretable.

In the future, we will also introduce various motivational and persuasion strategies that will help instructors in performing timely interventions and providing needed feedback. Motivational and persuasion techniques will help online students in improving their study behavior and reduce students' dropouts.

REFERENCES

[1] S. Palvia, P. Aeron, P. Gupta, D. Mahapatra, R. Parida, R. Rosner, and S. Sindhi, "Online education: Worldwide status, challenges, trends, and implications," *J. Global Inf. Technol. Manag.*, vol. 21, no. 4, pp. 233–241, 2018.

[2] S. Balkaya and U. Akkucuk, "Adoption and use of learning management systems in education: The role of playfulness and self-management," *Sustainability*, vol. 13, no. 3, p. 1127, Jan. 2021.

[3] C. Romero and S. Ventura, "Educational data mining and learning analytics: An updated survey," *WIREs Data Mining Knowl. Discovery*, vol. 10, no. 3, p. e1355, May 2020.

[4] S. Gaftandzhieva, A. Talukder, N. Gohain, S. Hussain, P. Theodorou, Y. K. Salal, and R. Doneva, "Exploring online activities to predict the final grade of student," *Mathematics*, vol. 10, no. 20, p. 3758, Oct. 2022.

[5] M. Hussain, S. Hussain, W. Zhang, W. Zhu, P. Theodorou, and S. M. R. Abidi, "Mining Moodle data to detect the inactive and low-performance students during the Moodle course," in *Proc. 2nd Int. Conf. Big Data Res. (ICBDR)*, 2018, pp. 133–140.

[6] M. Adnan, A. Habib, J. Ashraf, S. Mussadiq, A. A. Raza, M. Abid, M. Bashir, and S. U. Khan, "Predicting at-risk students at different percentages of course length for early intervention using machine learning models," *IEEE Access*, vol. 9, pp. 7519–7539, 2021.

[7] D. M. Ahmed, A. M. Abdulazeez, D. Q. Zeebaree, and F. Y. H. Ahmed, "Predicting university's students performance based on machine learning techniques," in *Proc. IEEE Int. Conf. Autom. Control Intell. Syst. (I2CACIS)*, Jun. 2021, pp. 276–281.

[8] H. S. Alenezi and M. H. Faisal, "Utilizing crowdsourcing and machine learning in education: Literature review," *Educ. Inf. Technol.*, vol. 25, no. 4, pp. 2971–2986, Jul. 2020.

[9] L. Hui and T. Chin-Chung, "A review of using machine learning approaches for precision education," *J. Educ. Technol. Soc.*, vol. 24, no. 1, pp. 250–266, 2021.

[10] A. A. Mubarak, H. Cao, and W. Zhang, "Prediction of students' early dropout based on their interaction logs in online learning environment," *Interact. Learn. Environments*, vol. 30, no. 8, pp. 1–20, 2020.

[11] A. S. Imran, F. Dalipi, and Z. Kastrati, "Predicting student dropout in a MOOC: An evaluation of a deep neural network model," in *Proc. 5th Int. Conf. Comput. Artif. Intell. (ICCAI)*, 2019, pp. 190–195.

[12] E. Alqurashi, "Predicting student satisfaction and perceived learning within online learning environments," *Distance Educ.*, vol. 40, no. 1, pp. 133–148, Jan. 2019.

[13] A. W. Cole, L. Lennon, and N. L. Weber, "Student perceptions of online active learning practices and online learning climate predict online course engagement," *Interact. Learn. Environments*, vol. 29, no. 5, pp. 866–880, Jul. 2021.

[14] D. Shin, Y. Shim, H. Yu, S. Lee, B. Kim, and Y. Choi, "SAINT+: Integrating temporal features for EdNet correctness prediction," in *Proc. 11th Int. Learn. Analytics Knowl. Conf.*, Apr. 2021, pp. 490–496.

[15] B.-H. Kim, E. Vizitei, and V. Ganapathi, "GritNet: Student performance prediction with deep learning," 2018, *arXiv:1804.07405*.

[16] O. H. T. Lu, A. Y. Q. Huang, J. C. H. Huang, A. J. Q. Lin, H. Ogata, and S. J. H. Yang, "Applying learning analytics for the early prediction of students' academic performance in blended learning," *J. Educ. Technol. Soc.*, vol. 21, no. 2, pp. 220–232, 2018.

[17] R. Hasan, S. Palaniappan, A. R. A. Raziff, S. Mahmood, and K. U. Sarker, "Student academic performance prediction by using decision tree algorithm," in *Proc. 4th Int. Conf. Comput. Inf. Sci. (ICCOINS)*, Aug. 2018, pp. 1–5.

[18] G. Casalino, C. Castiello, N. Del Buono, F. Esposito, and C. Mencar, "Q-matrix extraction from real response data using nonnegative matrix factorizations," in *Proc. Int. Conf. Comput. Sci. Appl.* Cham, Switzerland: Springer, 2017, pp. 203–216.

[19] R. Ghorbani and R. Ghousi, "Comparing different resampling methods in predicting students' performance using machine learning techniques," *IEEE Access*, vol. 8, pp. 67899–67911, 2020.

[20] B. Sekeroglu, K. Dimililer, and K. Tuncal, "Student performance prediction and classification using machine learning algorithms," in *Proc. 8th Int. Conf. Educ. Inf. Technol.*, Mar. 2019, pp. 7–11.

[21] D. Gunning and D. Aha, "DARPA's explainable artificial intelligence (XAI) program," *AI Mag.*, vol. 40, no. 2, pp. 44–58, 2019.

[22] N. Mehdiyev and P. Fettke, "Explainable artificial intelligence for process mining: A general overview and application of a novel local explanation approach for predictive process monitoring," in *Interpretable Artificial Intelligence: A Perspective of Granular Computing*. Ithaca, NY, USA: Cornell Univ., 2021, pp. 1–28.

[23] D. Minh, H. X. Wang, Y. F. Li, and T. N. Nguyen, "Explainable artificial intelligence: A comprehensive review," *Artif. Intell. Rev.*, vol. 55, no. 5, pp. 1–66, 2021.

[24] L. Longo, R. Goebel, F. Lecue, P. Kieseberg, and A. Holzinger, "Explainable artificial intelligence: Concepts, applications, research challenges and visions," in *Proc. Int. Cross-Domain Conf. Mach. Learn. Knowl. Extraction*. Cham, Switzerland: Springer, 2020, pp. 1–16.

[25] I. Ahmed, G. Jeon, and F. Piccialli, "From artificial intelligence to explainable artificial intelligence in industry 4.0: A survey on what, how, and where," *IEEE Trans. Ind. Informat.*, vol. 18, no. 8, pp. 5031–5042, Aug. 2022.

[26] O. Loyola-Gonzalez, "Black-box vs. white-box: Understanding their advantages and weaknesses from a practical point of view," *IEEE Access*, vol. 7, pp. 154096–154113, 2019.

[27] A. Rawal, J. McCoy, D. B. Rawat, B. M. Sadler, and R. St. Amant, "Recent advances in trustworthy explainable artificial intelligence: Status, challenges, and perspectives," *IEEE Trans. Artif. Intell.*, vol. 3, no. 6, pp. 852–866, Dec. 2022.

[28] M. Oussalah, "AI explainability. A bridge between machine vision and natural language processing," in *Proc. Int. Conf. Pattern Recognit.* Cham, Switzerland: Springer, 2021, pp. 257–273.

[29] K. A. Tarnowska, B. C. Dispoto, and J. Conragan, "Explainable AI-based clinical decision support system for hearing disorders," *AMIA Summits Transl. Sci. Proc.*, vol. 2021, p. 595, May 2021.

[30] M. Danilevsky, K. Qian, R. Aharonov, Y. Katsis, B. Kawas, and P. Sen, "A survey of the state of explainable AI for natural language processing," 2020, *arXiv:2010.00711*.

[31] D. Das, S. Banerjee, and S. Chernova, "Explainable AI for robot failures: Generating explanations that improve user assistance in fault recovery," in *Proc. ACM/IEEE Int. Conf. Hum.-Robot Interact.*, Mar. 2021, pp. 351–360.

[32] P. Madumal, T. Miller, L. Sonenberg, and F. Vetere, "A grounded interaction protocol for explainable artificial intelligence," 2019, *arXiv:1903.02409*.

[33] U. Ehsan and M. O. Riedl, "Human-centered explainable AI: Towards a reflective sociotechnical approach," in *Proc. Int. Conf. Hum.-Comput. Interact.* Cham, Switzerland: Springer, 2020, pp. 449–466.

[34] W. Zhang and B. Y. Lim, "Towards relatable explainable AI with the perceptual process," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, Apr. 2022, pp. 1–24.

[35] A. Holzinger, B. Malle, A. Saranti, and B. Pfeifer, "Towards multi-modal causability with graph neural networks enabling information fusion for explainable AI," *Inf. Fusion*, vol. 71, pp. 28–37, Jul. 2021.

[36] F. K. Došilović, M. Brčić, and N. Hlupić, "Explainable artificial intelligence: A survey," in *Proc. 41st Int. Conv. Inf. Commun. Technol., Electron. Microelectron. (MIPRO)*, 2018, pp. 0210–0215.

[37] S. Han, M. Kwon, and H. Choi, "EXplainable AI (XAI) approach to image captioning," *J. Eng.*, vol. 2020, no. 13, pp. 589–594, Jul. 2020.

[38] A. Rai, "Explainable AI: From black box to glass box," *J. Acad. Marketing Sci.*, vol. 48, no. 1, pp. 137–141, Jan. 2020.

[39] U. Pawar, D. O'Shea, S. Rea, and R. O'Reilly, "Explainable AI in healthcare," in *Proc. Int. Conf. Cyber Situational Awareness, Data Anal. Assessment (CyberSA)*, Jun. 2020, pp. 1–2.

[40] X.-H. Li, C. Chen Cao, Y. Shi, W. Bai, H. Gao, L. Qiu, C. Wang, Y. Gao, S. Zhang, X. Xue, and L. Chen, "A survey of data-driven and knowledge-aware eXplainable AI," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 1, pp. 29–49, Jan. 2022.
[41] O. Loyola-Gonzalez, A. E. Gutierrez-Rodriguez, M. A. Medina-Perez, R. Monroy, J. F. Martinez-Trinidad, J. A. Carrasco-Ochoa, and M. Garcia-Borroto, "An explainable artificial intelligence model for clustering numerical databases," *IEEE Access*, vol. 8, pp. 52370–52384, 2020.
[42] J. Tao, Y. Xiong, S. Zhao, R. Wu, X. Shen, T. Lyu, C. Fan, Z. Hu, S. Zhao, and G. Pan, "Explainable AI for cheating detection and churn prediction in online games," *IEEE Trans. Games*, vol. 99, p. 1, 2022.
[43] J. L. Zambrano, J. A. L. Torralbo, and R. Morales, "Early prediction of student learning performance through data mining: A systematic review," *Psicothema*, vol. 33, no. 3, pp. 456–465, 2021.
[44] H. Hagras, "Toward human-understandable, explainable AI," *Computer*, vol. 51, no. 9, pp. 28–36, Sep. 2018.
[45] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, "From local explanations to global understanding with explainable AI for trees," *Nature Mach. Intell.*, vol. 2, no. 1, pp. 56–67, 2020.
[46] R. Ding, W. Yin, G. Cheng, Y. Chen, J. Wang, R. Wang, Z. Rui, J. Li, and J. Liu, "Boosting the optimization of membrane electrode assembly in proton exchange membrane fuel cells guided by explainable artificial intelligence," *Energy AI*, vol. 5, Sep. 2021, Art. no. 100098.
[47] G. Visani, E. Bagli, F. Chesani, A. Poluzzi, and D. Capuzzo, "Statistical stability indices for LIME: Obtaining reliable explanations for machine learning models," *J. Oper. Res. Soc.*, vol. 73, no. 1, pp. 91–101, Jan. 2022.
[48] G. Montavon, A. Binder, S. Lapuschkin, W. Samek, and K.-R. Müller, "Layer-wise relevance propagation: An overview," in *Explainable AI: Interpreting, Explaining Visualizing Deep Learning*. 2019, pp. 193–209.
[49] T. Verma, C. Lingenfelder, and D. Klakow, "Explaining black-box predictions by generating local meaningful perturbations," *Int. J. Semantic Comput.*, vol. 16, no. 1, pp. 47–68, Mar. 2022.
[50] S. Tasaki, C. Gaiteri, S. Mostafavi, and Y. Wang, "Deep learning decodes the principles of differential gene expression," *Nature Mach. Intell.*, vol. 2, no. 7, pp. 376–386, Jul. 2020.
[51] M. Vega García and J. L. Aznarte, "Shapley additive explanations for NO_2 forecasting," *Ecol. Informat.*, vol. 56, Mar. 2020, Art. no. 101039.
[52] H. Wan, K. Liu, Q. Yu, and X. Gao, "Pedagogical intervention practices: Improving learning engagement based on early prediction," *IEEE Trans. Learn. Technol.*, vol. 12, no. 2, pp. 278–289, Apr. 2019.
[53] T. Soffer and A. Cohen, "Students' engagement characteristics predict success and completion of online courses," *J. Comput. Assist. Learn.*, vol. 35, no. 3, pp. 378–389, Jun. 2019.
[54] A. Kaur, A. Mustafa, L. Mehta, and A. Dhall, "Prediction and localization of student engagement in the wild," in *Digital Image Computing: Techniques and Applications (DICTA)*. Ithaca, NY, USA: Cornell Univ., 2018, pp. 1–8.
[55] J. L. Rastrollo-Guerrero, J. A. Gómez-Pulido, and A. Durán-Domínguez, "Analyzing and predicting students' performance by means of machine learning: A review," *Appl. Sci.*, vol. 10, no. 3, p. 1042, 2020.
[56] I. E. Livieris, K. Drakopoulou, V. T. Tampakas, T. A. Mikropoulos, and P. Pintelas, "Predicting secondary school students' performance utilizing a semi-supervised learning approach," *J. Educ. Comput. Res.*, vol. 57, no. 2, pp. 448–470, Apr. 2019.
[57] X. Xu, J. Wang, H. Peng, and R. Wu, "Prediction of academic performance associated with internet usage behaviors using machine learning algorithms," *J. Comput. Hum. Behav.*, vol. 98, pp. 166–173, Sep. 2019.
[58] I. M. K. Ho, K. Y. Cheong, and A. Weldon, "Predicting student satisfaction of emergency remote learning in higher education during COVID-19 using machine learning techniques," *PLoS ONE*, vol. 16, no. 4, Apr. 2021, Art. no. e0249423.
[59] C. C. Gray and D. Perkins, "Utilizing early engagement and machine learning to predict student outcomes," *Comput. Educ.*, vol. 131, pp. 22–32, Apr. 2019.
[60] B. K. Daniel, "Big data and data science: A critical review of issues for educational research," *Brit. J. Educ. Technol.*, vol. 50, no. 1, pp. 101–113, Jan. 2019.
[61] C. Fischer, Z. A. Pardos, R. S. Baker, J. J. Williams, P. Smyth, R. Yu, S. Slater, R. Baker, and M. Warschauer, "Mining big data in education: Affordances and challenges," *Rev. Res. Educ.*, vol. 44, no. 1, pp. 130–160, Mar. 2020.
[62] F. Chen and Y. Cui, "Utilizing student time series behaviour in learning management systems for early prediction of course performance," *J. Learn. Anal.*, vol. 7, no. 2, pp. 1–17, Sep. 2020.
[63] D. Baneres, M. E. Rodriguez, and M. Serra, "An early feedback prediction system for learners at-risk within a first-year higher education course," *IEEE Trans. Learn. Technol.*, vol. 12, no. 2, pp. 249–263, Apr. 2019.
[64] G. Kostopoulos, T. Panagiotakopoulos, S. Kotsiantis, C. Pierrakeas, and A. Kameas, "Interpretable models for early prediction of certification in MOOCs: A case study on a MOOC for smart city professionals," *IEEE Access*, vol. 9, pp. 165881–165891, 2021.
[65] S. Alwarthan, N. Aslam, and I. U. Khan, "An explainable model for identifying at-risk student at higher education," *IEEE Access*, vol. 10, pp. 107649–107668, 2022.
[66] S. Karlos, G. Kostopoulos, and S. Kotsiantis, "Predicting and interpreting students' grades in distance higher education through a semi-regression method," *Appl. Sci.*, vol. 10, no. 23, p. 8413, Nov. 2020.

MUHAMMAD ADNAN received the bachelor's degree in computer science from the University of Peshawar, the master's degree in information technology from the School of Electrical Engineering and Computer Science, National University of Science and Technology, (NUST), Islamabad, and the Ph.D. degree in machine learning and deep learning from the Institute of Computing, KUST, in 2021. His research interests include mobile learning, machine learning, adaptive learning, intelligent learning systems, and ubiquitous systems.

M. IRFAN UDDIN is currently working as an Assistant Professor with the Institute of Computing, Kohat University of Science and Technology (KUST), Pakistan. He has 15 years of teaching and research experience. His research interests include fuzzy logic, transformers, convolutional neural networks, and image processing.

EMEL KHAN received the Ph.D. degree in mathematics from the NJIT, USA. He is currently working as an Assistant Professor with the Institute of the Numerical Science, Kohat University of Science and Technology, Kohat. His research interests include mathematical modeling, optimization, numerical analysis, and machine learning.

FAHD S. ALHARITHI received the bachelor's degree in computer science from Taif University (TU), Saudi Arabia, in 2008, the master's degree in computer science from the University of New Haven (UNH), USA, in 2012, and the Ph.D. degree in computer science from the Florida Institute of Technology, USA, in 2019. He is currently an Assistance Professor with the College of Computers and Information Technology, TU. His research interests include human–computer interaction, cloud computing, the Internet of Things, artificial intelligence, and machine learning.

SAMINA AMIN is currently pursuing the Ph.D. degree in computer science from the Institute of Computing, Kohat University of Science and Technology, Kohat. Her research interests include reinforcement learning, machine learning, and deep learning.

AHMAD A. ALZAHRANI is currently working as a Senior Faculty Member with the Department of Information Systems, College of Computers and Information Systems, Umm Al-Qura University, Saudi Arabia. His research interests include artificial intelligence, data mining, and machine learning.