Predicting Student Performance from Online Engagement Activities Using Novel Statistical Features

Ghassen Ben Brahim¹

Received: 25 August 2021 / Accepted: 26 December 2021 / Published online: 17 January 2022
© King Fahd University of Petroleum & Minerals 2022

Abstract

Predicting students' performance during their years of academic study has been investigated tremendously. It offers important insights that can help and guide institutions to make timely decisions and changes leading to better student outcome achievements. In the post-COVID-19 pandemic era, the adoption of e-learning has gained momentum and has increased the availability of online related learning data. This has encouraged researchers to develop machine learning (ML)-based models to predict students' performance during online classes. The study presented in this paper, focuses on predicting student performance during a series of online interactive sessions by considering a dataset collected using digital electronics education and design suite. The dataset tracks the interaction of students during online lab work in terms of text editing, a number of keystrokes, time spent in each activity, etc., along with the exam score achieved per session. Our proposed prediction model consists of extracting a total of 86 novel statistical features, which were semantically categorized in three broad categories based on different criteria: (1) activity type, (2) timing statistics, and (3) peripheral activity count. This set of features were further reduced during the feature selection phase and only influential features were retained for training purposes. Our proposed ML model aims to predict whether a student's performance will be low or high. Five popular classifiers were used in our study, namely: random forest (RF), support vector machine, Naïve Bayes, logistic regression, and multilayer perceptron. We evaluated our model under three different scenarios: (1) 80:20 random data split for training and testing, (2) fivefold cross-validation, and (3) train the model on all sessions but one which will be used for testing. Results showed that our model achieved the best classification accuracy performance of 97.4% with the RF classifier. We demonstrated that, under similar experimental setup, our model outperformed other existing studies.

Keywords Machine learning · Random forest · Student performance prediction · Feature extraction · Binary classification

1 Introduction

The prediction of student performance has been the focus of many educational institutions due to the insight that it provides in terms of dropouts and learning outcome attainment. Such insights may support institutions in making learning adjustments or adopting new learning management strategies to create novel learning opportunities for the students. With the application of technology in the learning process through the wide deployment of learning management systems (LMS) and course learning platforms, the amount of

collected data is becoming significantly large to warrant sophisticated and intelligent techniques to manage and analyze such data.

Student performance prediction is considered an important area of educational data mining (EDM) and Learning Analytics (LA) [1]. It is becoming more challenging in the presence of the ever-increasing data volumes. EDM and LA are considered to be closely related disciplines and they aim to support the process of analyzing related educational data. Together, they provide tools and techniques to collect, process, and analyze educational data. In most cases, the analysis is based on data mining, machine learning, and statistical analysis techniques to explore hitherto unknown patterns in different types of historical data.

Researchers have focused on different data features to predict student performance: these include previous exam scores, background, demographic data, and activity in and

✉ Ghassen Ben Brahim
gbrahim@pmu.edu.sa

¹ Department of Computer Science, College of Computer Engineering and Science, Prince Mohammad Bin Fahd University, Al-Khobar 31952, Saudi Arabia

outside the classroom, to name but a few [2]. Though most of the performance prediction work tends to focus on previous exam scores, very little work seems to target the analysis of data wherein student interactions with online systems is being logged and analyzed [3]. Using past scores has two disadvantages: one is that it predicts performance in the long term such as predicting students' performance in his junior or sophomore year courses based on exam grades and course grades from the freshman or sophomore year, second is that even if it takes grades of major1 and major2 exams to predict the performance in the course, then it is too far in the semester for any corrective actions to be taken to assist the student to succeed. Therefore, in the work introduced herein, we propose focusing on performance prediction problems while exploring data describing students' online interactions during the online exam sessions. We experiment with a dataset that has been collected using digital electronics education and design suite (DEEDS)—a simulation software used for e-learning in a Computer Engineering course (Digital Electronics) [1]. The DEEDS platform logs student activities and actions during exams (such as viewing/studying exam content, working on a specific exercise, using a text editor or IDE, or viewing exam-related material). Our literature survey indicates that this DEEDS dataset was explored twice: in [1, 4], wherein authors attempted to predict student performance based on exam complexity and predict the exam difficulty based on student activities from prior sessions, respectively.

The aim of this research work is to build a prediction model which is based on newly extracted statistical features aimed at predicting students' performance based on their online activities. To build and refine the model, we have proceeded as follows. Initially, we have proposed new features that were categorized into three broad categories, based on different criteria: (1) Activity-type count-based, (2) Timing statistics-based, and (3) Peripheral activity count-based, resulting in a total of 86 features. We have also proposed further improvement in the model by reducing the set of features and eventually keeping the most influential (significant) ones using the entropy-based feature selection method [35, 36]. The proposed model was then evaluated and compared with other existing similar research work. We compared the performance with some of the existing work addressing the same problem and using the same DEEDS dataset. We have shown that our proposed model outperforms existing ones in terms of classification accuracy results.

The key contributions of this research work can be summarized as follows.

- Categorization of student academic performance-related features in existing datasets.
- Statistical analysis of the DEEDS dataset which supported the feature extraction process.

- The design of a student performance prediction model based on the extraction of a set of statistical features which were categorized into three broad categories: (1) activity-type based, (2) timing statistics-based, and (3) peripheral activity count-based. These features comprise an extraction phase followed by a feature selection phase using an entropy-based selection method.
- Performance evaluation of the proposed model by considering the following classifiers: random forest (RF), support vector machine (SVM), Naïve Bayes (NB), logistic regression (LR), and multilayer perceptron (MLP).
- Comparative performance analysis between our proposed model and some of the existing published research proposing students' performance prediction models using the DEEDS dataset [4, 30, 31].

The rest of the paper is organized as follows. Section 2 presents the background about the student performance prediction domain: its importance, applicable prediction metrics, dataset categorization, and overview of prediction models. Section 3 describes our proposed approach considering student engagement data wherein the DEEDS dataset is presented along with the feature extraction process. Section 4, details the model performance in terms of prediction accuracy, and then a comparative study is presented in relation to the existing work. Finally, in Sect. 5, conclusions are drawn, and future research directions are suggested.

2 Student Performance Prediction: Overview

This section starts with a brief background about the importance of student performance prediction. Then it overviews the performance prediction targets in terms of prediction goals. Next, it surveys and categorizes the set of features commonly considered in most of the datasets used during the prediction process. Finally, it overviews the prominent approaches and models being used along with their achieved performance.

2.1 Importance of Student Performance Prediction

The problem of predicting student performance has been extensively studied by the research community as part of the learning analytics topic due to its importance for many academic disciplines [2, 3, 5–7]. Based on the goal of the performance prediction model, the benefits may include the following: (1) improved planning and accurate adjustments in education management strategies to yield enhanced attainment rates in program learning outcomes [8], (2) identify, track and improve student learning outcomes and their impact on classroom activities. For instance, prediction models could be tuned to classify student performance as low,

average, or high. Based on the classification results, concerted measures may be taken by the education managers to support the low-performing students [7], (3) propose new, formative learning approaches for the students based on their predicted performance [8]. For instance, students are advised to adopt different learning strategies such as emphasizing more on practical aspects of course material, (4) allocating resources to the students based on their predicted performance. For instance, the identification and prediction of high-performing students will support institutions to estimate the number of awarded scholarships [9], (5) minimize the student dropout rates which is considered a resources black hole that impacts graduation rates, quality, and even institutional ranking [10].

2.2 What to Predict?

Student performance prediction models have targeted several metrics which are both quantitative and qualitative in nature. The amount of research work to predict quantitative metrics outweighs those for qualitative metrics [8]. Qualitative metrics have mainly focused on Pass/Fail or Letter Grade classifications of students in particular courses [11] or overall student assessment prediction in terms of high/average/low. This type of assessment could be performed per course, major, topic, etc. [3], or student knowledge accomplishment levels: First/Second/Third/Fail [4, 12], or to classify students into low risk/high risk/medium risk [7]. By contrast, quantitative metrics have mainly attempted to predict scores or course/exam/assignment grades [5], range of course/exam/assignment grades [6], major dropout/retention rates [10], prediction of the time needed for exam completion, prediction of on-duration/delay of graduation and student engagement as well [13].

2.3 Dataset Features Categorizations

Most of the datasets that have been used to machine-learn the student performance have considered historic data that can be categorized into three broad categories based on the attribute types [2]: (1) student historic performance attributes, (2) student demographic attributes, and (3) student learning platform interactions (engagement) attributes. These categorizations were further extended into a more comprehensive classification of features to include two more categories [3], namely: (4) personality—to better describe the subject capability and ability (such as efficacy, commitment, efficiency, etc.), and (5) institutional—to better describe the teaching methods, strategies, and qualities [14]. We have surveyed several datasets that have been considered for student performance prediction studies and have summarized them in Table 1, where we enlisted the five categories along with most of the common and relevant features being used in each

of these five categories. Table 1 also includes the stated aim of the prediction study per category.

2.4 Related Work

Several methods and approaches were considered in predicting student performance; most of these approaches are statistical in nature and designed for machine learning (ML) models. The models attempt to estimate an inherent correlation between input variables and identify patterns within the input data. Following our review of most of the existing datasets, these attributes can be classified under any of five categories, namely: (1) student historic performance attributes, (2) student demographic attributes, and (3) student learning platform interactions attributes, (4) personality attributes, and (5) institutional attributes; as detailed in Table 1.

Among the two existing ML models types, supervised learning techniques are a better fit for handling classification and regression problems and were more widely used to deal with the student prediction problem as compared to the unsupervised learning techniques. Classification approaches attempt to classify entities into some known classes, which are two in the case of binary classification (for example, classifying students into Passing or Failing classes) or more in the case of multinomial classification. On the other hand, in regression-based approaches, the model attempts to predict a continuous type of value (for instance, predict the final exam score, which could be a real number between “0 to 100”). This makes regression techniques more challenging problems to solve compared to the classification problems.

In the context of student performance prediction, several supervised learning models were used while considering datasets with each of the five feature categories as well as their combinations and targeting the prediction of specific performance features (as specified in Table 1). For instance, the authors in [2] have studied the impact of each of the three categories of features (student engagement, demographics, and performance data) on predicting student performance using binary classification-based models to predict at-risk students and regression techniques to predict the student scores. Also, they studied the prediction performance at different time instances before taking the final exam. The analysis was performed on a public open university learning analytics dataset (OULAD) while using Support vector machine (SVM), decision tree (DT), artificial neural networks (ANNs), Naïve Bayes (NB), K-nearest neighbor (K-NN), and logistic regression (LR) models for classification and SVM, ANN, DT, Bayesian Regression (BN), K-NN, and Linear Regression models for regression analysis. For the classification task, better performance results were obtained with ANN while considering engagement and exam score data (F1-score ~ 96%). The same performance was also

Table 1 Summary of features categories v/s targeted attributes

Feature categories	Features sub-categories	Features	Targeted/predicted attributes
Historic performance [2]	Pre-course performance	Exam/assignment/ project/lab/quiz/seminars results, previous semester performance, Admission score, cumulative GPA, topic performance, average course grade, course Pass/Fail	Course/Program dropout & retention, assessment grades, GPA range, actual GPA, course Pass/Fail, course letter grade, course grade, assignment grade
	Course performance		
Demographic [27]	Prior school performance		
	Family background Working conditions Population	Gender, age, marital status, children, occupation, computer literacy, skills, location, income, caretaker, support, ethnicity, international, minority, distance to school, daily commute time, internet access	Program dropout & retention, GPA range, course Pass/Fail, course letter grade, assignment grade
Learning platform interactions (engagement) [1]	Time-based performance	Duration/number of visited material, search activity, face-to-face meetings, discussions participation, frequency of deck comments, number of attempted exams, number of clicks, attendance, chat/forum activity, time on task, system interaction duration	Assessment grades, course Pass/Fail, course letter grade, course grade, assignment grade
Personality [3]	Activity count interaction performance		
	Studying strategy based	Efficacy, motivation, study habits, anxiety, positive working strategy, negative working strategy, goal-oriented, emotional, learning style	GPA range, course Pass/Fail, course letter grade, program dropout & retention
Institutional [14]	Feeling based		
	Teaching strategy Institution type	University educations, high school, pedagogical mode, teaching mode, formal/informal education, training mode, type of material available, deep/shallow learning approach	Course/Program dropout & retention, at-risk students

obtained for the regression analysis task where ANN outperformed other algorithms while inputting the model with more historic assessment scores (RMSE ~ 14.59).

The authors in [5] have focused on predicting student performance based on historic exam grades and in-progress course exam grades. The goal of such a study is to identify, per area and per subject, the “at-risk” students and hence provide real-time feedback about the current status of the students. Such feedback would drive the appropriate remedial strategies to support these students and eventually help improve retention rates. The authors have conducted their studies on 335 students and 6358 assessment grades while considering 68 subjects categorized into 7 different knowledge areas. The prediction model used the Decision Tree algorithm to classify students into passing and failing categories. In their effort to identify the most influential variable, the authors run their model while considering all possible combinations of Final grades and their weighted Partial (1 to 3) ones. The best model accuracy performance was reported (96.5%) when all Partial grades were included in the prediction.

In a different study, Huang and Fang [7] focused on predicting student performance in a specific field: Engineering dynamics. The authors conducted their research while considering four predictive models (multiple linear regression (MLR), multilayer perceptron network, radial basis function network, and the support vector machine-based model). Their study aimed to identify the best prediction model and the most influential variable among a set of six considered ones leading to a more accurate prediction model. The dataset being considered in this study included only the historic performance type of data. The dataset was a collection of 323 students over 4 semesters and included nine different grade dynamics and pre-requisite dynamic courses. Results showed that the type of mathematical model being used had little impact on the prediction accuracy. Regarding the most influential variables, results showed that it varies based on the goal of the instructor in terms of what to predict (average performance of the entire dynamic class or individual student performance). Best performance results in terms of accuracy were achieved with MLR (reaching 89.7%). Similar to the proposed model in [5], this model only considered one category of student data (historic performance-related data) and did not include different categories such as engagement or demographic data.

As online teaching is gaining more and more popularity, it is becoming necessary for all schools to provide e-learning options to their students, especially after the COVID-19 pandemic. Many research works have studied and evaluated the performance of such learning in a virtual environment. For instance, Hussain et al. [15] have focused on studying the impact of student engagement in a virtual learning environment on their performance in terms of attained exam scores. This study has considered various variables including demographics, assessment scores, and student-system

engagements such as the number of clicks that student executes to access certain URLs, resources, homepages, etc. The authors have considered several machine learning classification algorithms to predict low engagement instances, such as Decision Trees (DT), JRIP, J48, gradient-boosted tree (GBT), CART, and Naïve Bayes. The best performance in terms of predicting students with low engagement was obtained with the first four algorithms (topping 88.5%). Also, these results have identified the best predictive variables in terms of the number of clicks students executed per activity during the identification of low-engagement students.

In the same context, Vahdat et al. [1] aimed to study the impact of student behavior during the online assessment and the scores obtained. The authors have used complexity matrix and process mining techniques to identify any correlation between attained scores and student engagement. The dataset was collected using digital electronics education and design suite (DEEDS), a simulation software used for e-learning in a Computer Engineering course, namely Digital Electronics. Analysis has shown that: (1) the complexity matrix and student scores are positively correlated, (2) and the complex matrix and session difficulties are negatively correlated. Additionally, the authors demonstrated that the proposed process discovery could provide useful information about student learning practices.

In a different study Elbadrawy et al. [16], the authors have built a model to accurately predict student grades based on several types of features (past grade performance, course characteristics, and student interaction with the online learning management system (aka. Student Engagement)). The built model relies on a weighted sum of collaborative multi-regression-based models, which were able to improve the prediction accuracy performance by over 20%.

Along the same directions as in [16], Liu and d’aguin [17] have attempted to predict student performance based on two categories of features: Demographics and Student Engagement with the online learning system. They have applied supervised learning-based algorithms in their model on the Open University Learning Analytics dataset [18] and investigated the relationship between demographic features and the achieved performance. Analysis has shown that the best-performing students were those who had acquired a higher education level and were residing in the most privileged areas.

Hussain et al. in [4] have investigated predicting difficulties that students may face during Digital Electronics lab sessions based on previous student activities. They identified the best predicting model, as well as the most influential features. The authors have only considered the engagement type of data using the digital electronics education and design suite (DEEDS) simulator [1]. They have conducted their study considering the following five features: average time, average idle time, and the total number of activities, total related activ-

ity, and the average number of keystrokes. Five classification algorithms were explored: support vector machines (SVMs), artificial neural networks (ANNs), Naïve Bayes classifiers, Logistic Regression, and Decision Trees. While considering fivefold cross-validation and random data division, the best accuracy performance results were obtained with the ANN and SVM-based models (75%). This performance was later improved and reached 80% when considering the Alpha Investing technique on the SVM-based model.

DEEDS dataset has also been used in other research work where researchers have attempted to predict the performance of students and the difficulties they face by analyzing students' behavior during interactive online learning sessions. For instance, in [30], the authors attempted to predict the student performance using the DEEDS dataset by considering regression models. The input variables of the model were the current study of the students for all sessions, however, the model output consists of the student's grade for a specific session. Among the three models used, (Linear regression, Artificial Neural Networks, and Support Vector Machine (SVM)), SVM performed the best and achieved an accuracy of 95%.

In a different research work, the authors in [31] have also considered the DEEDS dataset to perform a comparative analysis using various machine learning models such as Artificial Neural Network, Logistic regression, Decision Tree, Support Vector Machines, and Naïve Bayes. In their study, the authors have extracted and considered five different types of features, including average time, total activities, average mouse clicks, related activities in an exercise, average keystrokes, and average idle time per exercise. During this study, SVM performed the best as compared to the rest of the models and has achieved an accuracy of 94%.

In [32], the authors have considered different datasets in their attempt to predict the performance of students in two different courses (namely Mathematics and Portuguese language). Both datasets have a total of 33 attributes each and a total of 396 and 649 records, respectively. The authors have applied two different types of models: following a feature Support Vector Machines and Random Forest to classify passing students from the failing ones. Both datasets comprise 16 features, including historic performance, demographic data, and personality type of features. Experimental results showed an accuracy of more than 91% and that the historic features were most influential during classification. It was also worth noting that Random Forest performed better in the case of larger dataset (Portuguese language).

The authors in [33] have proposed a model to predict students' performance in higher educational institutions using video learning analytics and data mining techniques. A sample of 722 students was considered in the collection of the dataset. Among the five categories highlighted in Table 1, only the historic performance, engagement, personality, and

institutional categories were used. Out of the eight algorithms used, the RF performed best and has achieved an accuracy of 88% following the feature reduction step.

In another study [34], the authors have used Artificial Neural Networks (ANN) to predict the performance of students in an online environment. The data set was collected using the participation of 3518 students. Out of the five categories of features described in Table 1, only historic performance and learning platform interaction categories were considered. Results showed that the ANN-based model was able to predict the students' performance with an accuracy of 80%. Summary of the prediction of student performance using machine learning-based models is presented in Table 2. It is worth noting that the last entry in Table 2 captures the performance achieved with the proposed model in predicting students' performance while considering student engagement and historic type of features.

Our proposed research explores the area of predicting student performance in an e-learning environment, which is gaining more popularity, especially post the COVID-19 pandemic. We propose exploring the DEEDS dataset, which is, to the best of our knowledge, studied twice by Vahdat et al. [1] and Hussain et al. [4]. DEEDS is a Technology Learning Environment platform that logs the real-time interactions of students during classwork as well as exam performance in terms of grades; these logs were collected in six different interactive sessions. We intend to conduct a statistical study on the DEEDS dataset followed by the design of a new prediction model based on a new set of statistical features to predict student performance using their interaction logs registered by DEEDS. We propose assessing our model's performance using five different types of classifiers in a different experimental setup. We will also compare the achieved performance in terms of accuracy and F1-score to that proposed by Hussain et al. [4].

3 Proposed Methodology

The proposed method aiming to classify student performance follows a typical machine learning classification approach. Initially, we start with a statistical analysis and feature engineering of the DEEDS dataset which resulted in the reduction of DEEDS activity features from 15 to 9 activities. This reduction (as will be described in Sect. 3.2) consists of aggregating semantically similar activities into a single category. The next step of this process consists of the data pre-processing phase, where entries showing some discrepancy were discarded, and DEEDS logs with no corresponding lab or final exam grades were also excluded. This is further explained in Sect. 3.3. The next step is a features extraction phase. This phase resulted in extracting three broad categories of features: (1) Activity-type count-based, (2)

Table 2 Summary of student performance prediction using machine learning-based models

References	Dataset source/description	Features categories	Models used	Top performance
[2]	Open University Learning Analytics (OULAD) dataset, 3166 students	Student engagement, demographics, historic performance	Support Vector Machine (SVM), Decision Tree (DT), Artificial Neural Networks (ANN), Naïve Bayes (NB), K-Nearest Neighbor (K-NN), and Logistic Regression (LR)	ANN (96% F1-score)
[5]	Ecuador University, 335 students	Historic performance	Decision Tree	96.5% accuracy
[7]	Engineering Dynamics course, 323 students	Historic performance	Multiple Linear Regression (MLR), Multilayer Perception (MLP) network, Radial Basis Function (RBF) network, Support Vector Machine (SVM)	MLR (89.7% accuracy)
[15]	OU University, United Kingdom, 384 students	Demographic, assessment scores, student-system engagement	Decision Tree (DT), JRIP, J48, Gradient-Boosted Tree (GBT), CART, and Naïve Bayes	J48 (88.5% accuracy)
[16]	University of Minnesota's Moodle, 11,556 students	Student engagement, historic performance	Multi-Regression	0.145 RMSE
[4]	DEEDS, 115 students	Student engagement, historic performance	Support Vector Machines (SVMs), Artificial Neural Networks (ANNs), Naïve Bayes classifiers, Logistic Regression, and Decision Trees	SVM (80% accuracy)
[30]	DEEDS, 115 students	Student engagement, historic performance	Linear Regression, Artificial Neural Networks, and Support Vector Machine (SVM)	SVM (95% accuracy)
[31]	DEEDS, 115 students	Student engagement, historic performance	Artificial Neural Network, Logistic Regression, Decision Tree, Support Vector Machine, and Naïve Bayes	SVM (94.8% accuracy)
[32]	Mathematics dataset, 696 records	Historic performance, demographic data, personality type of features	Support Vector Machine, Random Forest	SVM (92% accuracy)
[32]	Portuguese dataset, 649 records	Historic performance, demographic data, personality	Support Vector Machine, Random Forest	RF (94% accuracy)
[33]	722 students	Historic performance, engagement, personality, institutional	Neural Network, KNN, LR, CN2 Rule Inducer, Naïve Bayes, SVM	RF (88% accuracy)
[34]	3518 students	Only historic performance, student engagement	Artificial Neural Networks (ANN)	80%. accuracy
Proposed work	DEEDS, 115 students	Student engagement, historic performance	Random Forest (RF), Support Vector Machine (SVM), Naïve Bayes (NB), Logistic Regression (LR), and Multilayer Perceptron (MLP)	97.4% accuracy

Timing statistics-based, and (3) Peripheral activity count-based, resulting in a total of 86 features. This phase is described in Sect. 3.4. The final phase consists of applying classification algorithms including: Random Forest (RF), Support Vector Machines (SVM), Multi-Layer Perceptron (MLP), Naïve Bayes (NB), and Logistic Regression (LR). Our proposed model attempts to predict the following: weak versus good performing students based on interaction with DEEDS through the use of a binary classification models. The proposed model's performance was also compared to that proposed in [4], where the authors considered a set of 30 extracted features while considering the same DEEDS Dataset. Figure 1 shows a workflow diagram of the entire process that we followed to build our prediction model. The process involves the typical five machine learning steps, including: (1) DEEDS raw dataset collection, (2) DEEDS data logs pre-processing, (3) Features extraction, (4) Feature selection, and (5) classification and analysis.

3.1 Dataset Description

In this research work, DEEDS dataset was used. DEEDS is considered among the very few existing ones that capture real-time in-class student interactions and behavior while using a proprietary Enhanced Technology Learning (ETL) platform called DEEDS (Digital Electronics Education and Design Suite). This dataset was considered by this study due to each richness in terms of the variety of students interaction information being logged by the system such as: average time being spent by students in various types of problems as well as learning environments, types of activities, the total number of activities, average idle time, the average number of keystrokes and total related activities for various exercises during individual sessions, the grade performance achieved by students in each session, etc. Also, we claim that there is enough room for opportunity to develop a classification model which will achieve better performance compared to the existing ones, which is the main objective of this research work. General information about DEEDS is presented in Table 3.

DEEDS logs real-time interaction of students during class-work as well as exam performance in terms of grades. The DEEDS dataset was collected following six working lab sessions. Students will initially be learning a specific topic during each of these sessions, then taking a set of exercises. The number of exercises in each of the 6 sessions ranges from 4 to 6 exercises. For instance, in sessions 1, 3, and 5, four exercises were planned. In sessions 2 and 6, six exercises were planned. In session 3, five exercises were planned. Student performance (in terms of attained grades) is subsequently recorded; with the maximum session grade varying from 4 and 6. During a study session phase, DEEDS records all student activities in each session, such as using text edi-

Fig. 1 Workflow diagram of the proposed model for student performance prediction

tor, using a simulation timing diagram tool, reviewing study material, using the Aulaweb, etc. After attending all lab sessions, students will be taking an exam where questions are chosen to cover each of the six topics studied in each of the six lab sessions, and student performance per topic was also noted. DEEDS creates a separate file for each student attending a session and adds a new entry every one-second interval. Each new entry corresponds to a new row in these log files describing student activities during each session. Each row

Table 3 DEEDS dataset information

Owner	University of Genoa, Italy
Year	2015
Topic	Digital Electronics—Computer Engineering
Performance metric	Session grades, Final exam grades
Dataset Location	https://sites.google.com/site/learninganalyticsforall/data-sets/epm-dataset

Table 4 Dataset features description

Order	Feature	Description
1	Session	Session identifier (1.0.6)
2	Student	Student identifier (1.0.115)
3	Exercise	Exercise identifier (1.0.6)
4	Activity	Type of activity
5	Time begin	Activity Start time
6	Time end	Activity End time
7	Idle time	Idle time duration
8	Mouse wheel	Amount of mouse wheel during activity
9	Mouse Wheel click	Number of mouse clicks during activity
10	Mouse click left	Number of left mouse clicks during activity
11	Mouse click right	Number of right mouse clicks during activity
12	Mouse movement	Distance covered by mouse
13	Keystrokes	Keystrokes hit during activity

Table 5 Statistical analysis of the DEEDS dataset

DEEDS details	Statistics
Number of features	13
Number of lab sessions	6
Dataset size	230,318 entries
Number of activities	15
Number of students	115
Average number of students per session	86
Number of students not taking any test	7
Freq. (%) of non 0 entries for “mouse wheel”	21,460 (9.3%)
Freq. of non 0 entries for mouse wheel clicks	376 (0.1%)
Freq. of non 0 entries for mouse wheel click right	12,997 (5%)
Freq. of non 0 entries for mouse wheel click left	206,457 (90%)
Freq. of non 0 entries for mouse movement	215,576 (93%)
Freq. of non 0 entries for “Keystrokes”	33,706 (14%)

is a collection of 13 comma-separated features arranged in the order indicated in Table 4.

Figure 2 shows a snapshot of the comma-separated log file corresponding to student ID 21 collected during the 4th session.

3.2 Statistical Analysis of the DEEDS Dataset

As a result of the preliminary data pre-processing and *meta*-data analysis, we are able to better describe the dataset, as shown in Table 5. Though 115 students were expected to participate in this experiment and eventually take exams, 7 students ended up not showing up in all sessions, and an average of 86 students were registered per lab session. Our dataset included more than 230,000 entries, which were uniformly distributed across all sessions, as shown in Fig. 3. Sessions 1 through 5 included 13–17% of the entire dataset,

except for the last session, which included 23% of all data. Statistical analysis has also shown that some dataset features logged fewer activities, and most of the registered values were “0”. For the “number of mouse wheel” feature, 90.7% of the logged values were zeros’, in the “number of mouse wheel click” feature, 99.9% of the logged values were zeros’, in the “number of mouse wheel click right” feature, 95% of the logged values were zero, and 86% of logged “keystrokes” were zeros’. On the other hand, the number of “mouse wheel click left” and “mouse movement” features were well distributed across a range of 0 to 1096 and 0 to 85,945, respectively.

Our next analysis focused on the representation of each of the activities in the entire dataset. Some activities did not have sufficient representation throughout the entire dataset and were represented with less than 1%, occurrence such as “Text Editor no exercise—0.02%”, “Deeds no exercise—0.2%”, “Deeds other activity—0.4%”, “FSM related—0.1%”. This is in comparison with the rest of the activities, which had representation rates between 7.5 and 16.3% of the entire

Fig. 2 Raw data snapshot

```
4, 21, Es_4_1, Study_Es_4_1, 13.11.2014 11:8:17, 13.11.2014 11:12:51, 35205806, 5, 0, 2, 0, 94, 0
4, 21, Es_4_5, TextEditor_Es_4_5, 13.11.2014 11:50:49, 13.11.2014 11:50:57, 5986, 0, 0, 8, 0, 257, 6
4, 21, Es_4_5, Diagram, 13.11.2014 11:51:15, 13.11.2014 11:54:8, 89410, 0, 0, 149, 0, 6088, 0
4, 21, Es_4_5, Aulaweb, 13.11.2014 12:44:51, 13.11.2014 12:47:27, 10947165, 103, 0, 2, 0, 138, 0
```

Fig. 3 DEED's log distribution across sessions

Fig. 4 Frequency of each activity in the DEEDS dataset size (pre-aggregation)

dataset. Proportional activity frequency results are depicted in Fig. 4. To avoid dropping any of the least frequent activity data entries, we propose aggregating semantically equivalent activities under new activity categories. These are being represented in Fig. 5, along with their representation frequency. In this figure, activity “Editing” includes the following similar editing activities “Text Editor exercise”, “Text Editor no exercise”, and “Text Editor other activity” with an 18% frequency. The same applies for the new category “Deeds activity”, which includes “Deeds exercise”, “Deeds no exercise”, and “Deeds other activity” with a 17% overall frequency occurrence. The same applies to the new category “study”, which includes “study exercise” and “study material” with a 10% frequency. Also, new category “FSM” now includes activities “FSM exercise” and “FSM material” with a 9% frequency.

3.3 Data Pre-processing

Pre-processing is a typical first step in the data classification and pattern recognition process. During this phase,

data transformation and irrelevant information are being discarded from the original raw dataset, and only relevant data is kept for further processing. In the current work, the following steps have been executed on the original raw data.

(a) *Data discrepancy removal* We have noticed that the original dataset had some inconsistencies between the pairs (Session, Exercise) and (Exercise, Activity), as indicated in Fig. 5. DEEDS platform seems to have wrongly logged the session ID with its corresponding exercise ID (Fig. 6a) and Exercise ID with its corresponding Activity ID (Fig. 6b). The frequency of occurrences of such inconsistencies was minimal, and counts for less than 0.5% of the entire dataset and were proportionally equivalent across all sessions and exercises; therefore, these entries were discarded from the dataset. During this phase, we also excluded the entries where students have logged on but did not start working on the exercises. These are the logs with the exercise ID field matching “ES”.

(b) *Session 1 data removal* During this step, the session 1 data was filtered out. This was necessary since session 1 did not have any corresponding intermediate nor final exam grades related to the topics covered in this session. It is worth mentioning that student grades (intermediate or final exam) will be used to label our data for classification purposes.

(c) *Time format conversion* The DEEDS platform logs the time using the format “hours: minutes: seconds”. These times were converted to seconds to facilitate operations on these times.

(d) *Exercises 5 and 6 exclusion* This phase has to do with restricting our analysis to the first 4 exercises of each of the 5 sessions (2 through 6), the rest have been discarded. This step is necessary since not all sessions have the same number of exercises.

3.4 Features Extraction

This step is deemed pivotal during classification problems. In this work, we propose using an augmented set of numeric features compared to those by Hussain et al. (2019), where the authors have considered 30 features (5 features per exercise) for each student–session combination. We extracted a set of 86 features for each student per session and categorized these features into 3 broad categories: (1) *Activity-type* count-based features, (2) *Timing* statistics-based features, and (3) *peripheral activity* count-based features, which are explained below.

3.4.1 Activity-Type Count-Based Features

As stated in our dataset description, DEEDS defines 15 types of activities. Based on our statistical analysis in terms of log count distribution of each of these activities (Fig. 3) along with their significance, we propose reducing these lists

Fig. 5 Frequency of each activity in the DEEDS dataset size (post-aggregation)

Session	Student	Exercise	Activity	Start_T	End_T	Idle_T	mouse_w	mouse_whe	mouse_click	mouse_click	mouse_move	keystr
4	91	Es_5_2	TextEditor_Es_5_2	13.11.2014 11:37:41	13.11.2014 11:37:53	27673	6	0	2	0	252	0
4	91	Es_5_2	Other	13.11.2014 11:37:54	13.11.2014 11:37:59	343	0	0	4	0	171	0
6	82	Es_5_1	Other	11.12.2014 11:9:24	11.12.2014 11:9:28	96	0	0	8	0	509	0

(a): Session ID and Exercise ID

Sessio	Stude	Exerci	Activity	Start_T	End_T	Idle_T	mouse_w	mouse_wheel	mouse_click	mouse_click	mouse_move	keystro
2	87	Es_2_2	TextEditor_Es_3_2	16.10.2014 12:31:39	16.10.2014 12:31:41	814	0	0	3	0	127	0
2	99	Es_2_2	TextEditor_Es_4_2	16.10.2014 12:7:23	16.10.2014 12:7:25	124	0	0	5	0	59	0
2	99	Es_2_2	TextEditor_Es_4_1	16.10.2014 12:8:16	16.10.2014 12:13:56	41624944	0	0	17	0	479	513

(b): Exercise ID and Activity ID

Fig. 6 Example of DEEDS log discrepancy

by combining semantically similar non-proportional logs into a single group. In other words, activities {Deeds_Es_#, Deeds_Es, Deeds} are reduced into {Deeds Activity}, activities {TextEditor_Es_#, TextEditor_Es, TextEditor} are reduced into {Editing}, activity and activities {FSM_Es_#, FSM_Related} are reduced into {FSM}. This has resulted in the following Activity_type set of 9 categories {Editing, Aulaweb, Deeds Activity, FSM, Study, Blank, Other, Diagram, Properties}. Each of these 9 activities will be referenced with its order of appearance in the Activity_type set. For instance, activity 3 represents “Study” and activity 9 represents “Properties”.

Before we can mathematically express our activity-based features, we introduce the following activity matrix, which captures statistics about student activity occurrence count in all 4 exercises per session:

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \dots & a_{1,9} \\ a_{2,1} & a_{2,2} & a_{2,3} & \dots & a_{2,9} \\ a_{3,1} & a_{3,2} & a_{3,3} & \dots & a_{3,9} \\ a_{4,1} & a_{4,2} & a_{4,3} & \dots & a_{4,9} \end{bmatrix}$$

In matrix A , which is a 4 by 9 matrix, element “ $a_{i,j}$ ” represents the occurrence count of activity “ j ” in exercise “ i ”. For instance, “ $a_{1,3}$ ” represents the occurrence count of activity 3 (“Study”) in exercise 1.

Following the definition of matrix A , we define the first 36 features as depicted in Eq. 1 as follows:

$$F_{j=\{1 \rightarrow 36\}} = a_{[j/9], j-9[j/9]} \quad (1)$$

The 36 features map/track the occurrence count of the 9 activities in each of the 4 exercises (indexed by e). F_1 for instance = $a_{1,1}$ and represents the occurrence count of “Editing” activity in *exercise_1*.

For the next 9 features (F_{37} through F_{45}), we track the occurrence count of the 9 activities across all exercises. These are captured in Equations set 2. F_{37} , for instance, represents the occurrence count of “Editing” activity in all 4 exercises.

$$\begin{aligned} F_{37} &= \sum_{e=1 \rightarrow 4} a_{e,1}; \\ F_{38} &= \sum_{e=1 \rightarrow 4} a_{e,2} \ ; \dots \ ; \ F_{45} = \sum_{e=1 \rightarrow 4} a_{e,9} \end{aligned} \quad (2)$$

The next 4 features track the occurrence count of all 9 activities together in each of the 4 exercises. These are captured in Equations set 3. F_{46} , for instance, represents the aggregated occurrence count of all 9 activities in exercise 1.

$$\begin{aligned} F_{46} &= \sum_{i=1 \rightarrow 9} a_{1,i}; \quad F_{47} = \sum_{e=1 \rightarrow 4} a_{2,i}; \\ F_{48} &= \sum_{e=1 \rightarrow 4} a_{3,i}; \quad F_{49} = \sum_{e=1 \rightarrow 4} a_{4,i} \end{aligned} \quad (3)$$

The final feature in this activity count-based category tracks the aggregated sum of occurrence of all activities across all exercises. This is being captured in Eq. 4.

$$F_{50} = \sum_{i=1 \rightarrow 9} \sum_{e=1 \rightarrow 4} a_{e,i} \quad (4)$$

3.4.2 Timing Statistics Based Features

This category of features captures the timing performance of a student while working on exercises. We used 2 sets of timing features of size 4 features each. The first set of 4 features $\{F_{51}, F_{52}, F_{53}, F_{54}\}$ describes the time spent in each exercise. This is calculated by taking the difference of max-end time and min-start time for each of the 4 exercises as indicated in Eq. 5. F_{51} , for instance, corresponds to the period of time spent by a student in exercise 1.

$$F_{j=\{51 \rightarrow 54\}} = T_{\max_e} - T_{\min_e}; \quad \text{where } e = 1 \rightarrow 4 \quad (5)$$

The next set of 4 features ($F_{55}, F_{56}, F_{57}, F_{58}$) describes the total idle time registered in each of the 4 exercises. F_{55} , for instance, describes the total idle time registered in exercise 1. This is being captured in Eq. 6

$$F_{j=\{55 \rightarrow 58\}} = \sum_{1 \rightarrow n} Idle_Time_e; \quad \text{where } e = 1 \rightarrow 4 \quad (6)$$

3.4.3 Peripheral Activity Count-Based Features

The final set of features deals with tracking the count of student interactions with the computer input peripherals (mouse and keyboard). Since DEEDS logs 5 different types of mouse activities $\{MouseWheelCount,$

$MouseWheelClickCount,$ $MouseClickLeftcount,$ $MouseClickRightcount,$ $MouseMovementCount,$ $KeystrokeCount\}$ along with one single keyboard activity $\{KeystrokeCount\}$. We used 24 new features to track the total occurrence count of each of the aforementioned 6 peripheral related activities in each of the 4 exercises.

Each of these 6 peripheral activities will be mapped with its order of appearance in the Peripheral_Activity_type set. For instance, peripheral activity 3 represents “Mouse Click Left count” and activity 6 represents “Keystroke Count”.

Similar to the case of Activity-based features, we introduce the following peripheral activity matrix, which captures statistics about student interactions using the computer peripherals in all 4 exercises per session:

$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} & \dots & p_{1,6} \\ p_{2,1} & p_{2,2} & p_{2,3} & \dots & p_{2,6} \\ p_{3,1} & p_{3,2} & p_{3,3} & \dots & p_{3,6} \\ p_{4,1} & p_{4,2} & p_{4,3} & \dots & p_{4,6} \end{bmatrix}$$

In matrix P , which is a 4 by 6 matrix, element “ $p_{i,j}$ ” represents the occurrence count of peripheral activity “ j ” in exercise “ i ”. For instance, “ $p_{1,3}$ ” represents the occurrence count of peripheral activity 3 (“Mouse Click Left Count”) in exercise 1.

Each element of the peripheral activity-based matrix maps to a different feature, which will be considered during the classification phase. These constitute a set of 24 new features (F_{59} through F_{82}). F_{59} , for instance, describes the total occurrence count of *Mouse Wheel* events in exercise 1.

The next set of features within this category defines 4 more features to describe the level of utilization of the computer peripherals in each of the 4 exercises. These are described in equation set 7. F_{83} , for instance, specifies the total occurrence count of the 5 mouse activities and the keystrokes activities in exercise 1.

$$\begin{aligned} F_{83} &= \sum_{k=1 \rightarrow 6} p_{1,k}; \quad F_{84} = \sum_{k=1 \rightarrow 6} p_{2,k}; \\ F_{85} &= \sum_{k=1 \rightarrow 6} p_{3,k}; \quad F_{86} = \sum_{k=1 \rightarrow 6} p_{4,k} \end{aligned} \quad (7)$$

3.5 Feature Selection

Our feature selection phase consists of ranking all features by applying the entropy-based techniques based on the Shannon theory [35, 36]. This approach attempts to assign ranks based on the level of uncertainty or disorder within the data. Under this scheme, features with low entropy levels indicate a high level of uncertainty and, therefore, will be assigned low-rank values. The entropy-based technique is a powerful technique in determining the level of correlation between variables by capturing the linear and non-linear associations

between variables. We propose assessing the performance of our prediction model with the full set of extracted features and a reduced set after eliminating the none influential features (low ranked ones.)

3.6 Classification Algorithms

Among the many existing classification models, we have evaluated our model using five different classifiers. There are many factors influencing the suitability of the machine learning-based regression models, which may differ from one problem to another. These factors include the type of the data, the dataset size, the number of features, data distribution, etc. Also, models may work best for problems but not others. For instance, SVM is known to perform well in the case of relatively small dataset size, which is the case of DEEDS. In this work, models were chosen based on two criteria: (1) current existing work dealing with the same problems and using the same dataset DEEDS. For instance SVM, LR, NB. (2) cover the broad spectrum of the different categories of classifiers. For instance, RF was considered as a representative of the ensemble type of classifiers, NB as a probabilistic type of models, MLP as a Neural Network-based model. The model selection process also included a step where models which have experienced high error rates were eliminated.

Next, a brief description of each of the Classification models being used in this work is presented.

3.6.1 Random Forest (RF)

RF is an ensemble of Decision Tree bundled together [19]. The training of these bundles of trees consists of executing the bagging process on a dataset of N entities. This process consists of sampling a set of N training samples with replacement. Then using these samples to train a decision tree. This process needs to be repeated T times. The prediction of the unseen entity is eventually made through a majority vote of each of the T trees in the case of classification trees or the average value in the case of regression as given by Eq. 8.

$$y = \frac{1}{T} \sum_{i=1}^T f_i(x') \quad (8)$$

where y is the predicted value, x' is the unseen sample, f_i is the trained decision tree on data sample i , and T is a constant number of iterations to repeat the process.

$$y = \text{mode}\{f_1(x'), f_2(x'), \dots, f_i(x'), \dots, f_T(x')\} \quad (9)$$

where $f_i(x')$ is the prediction class of unseen entity x' using the decision tree trained on data sample i .

RF technique has shown its ability in handling large datasets with a large number of attributes while it weighs

the importance of each of the problem features. It is prone to noise, outliers, and overfitting [20]. Contrarily to other classification techniques, the RF relies on a combination of a set of classification techniques contributing to a single vote during the class classification process.

3.6.2 Multilayer Perceptron (MLP)

MLP is a supervised learning-based approach [21]. It is based on the concept of perceptron in Neural Networks, which is capable of generating a single output based on a multi-dimensional data input through their linear (non-linear in some occasions) combination along with their corresponding weight as follows:

$$y = \alpha \sum_{i=1}^n w_i x_i + \beta \quad (10)$$

where w_i , x_i , β , and α . are the weights, input variable, bias, non-linear activation function, respectively. The MLP is composed of three or more node layers, including the input/output layer and one or many hidden layers. The training phase in the case of MLP consists of adjusting the model parameters (biases and weights) through a back and forth mechanism (Feed-forward pass followed by Back-forward pass) with respect to the prediction error.

3.6.3 Support Vector Machine (SVM)

SVM is a supervised ML model to solve classification and regression problems; it has demonstrated efficiency in solving a variety of linear and non-linear problems. The idea of SVM lies in creating a hyperplane that distinctly categorizes the data into classes [22]. SVM works well for multi-domain applications with a large dataset; however, the model has a high computational cost.

3.6.4 Naïve Bayes (NB)

NB is a probabilistic algorithm based on Bayes' theorem. It is naïve in the sense that each feature makes an equal and independent contribution in determining the probability of the target class. NP has the advantage of noise immunity [23]. It is proven to perform well in the case of large high-dimensional datasets. It is fast (computational complexity-wise) and relatively easy to implement.

3.6.5 Logistic Regression (LR)

LR is an ML algorithm based on probability concepts used for classification—finding the success and failure events. LR can be considered as a Linear Regression model with a more complex cost function, which can be defined as the sigmoid

function (compared to a linear function in the case of linear regression [24]. LR has the advantage of being computationally efficient, relatively simple to implement with good performance with various types of problems. However, it has the main disadvantage of assuming linearity between independent and dependent variables [25].

4 Results and Analysis

4.1 Experiment Setup

Three sets of experiments aiming at three different goals were conducted during this study: (1) Goal_1—evaluate the performance of the proposed model while considering the full set of extracted features, (2) Goal_2—study the importance of each of the extracted features in the classification process using the entropy-based ranking approach, (3) Goal_3—compare the performance of our model to that proposed in [4, 30, 31].

The general experiment setup consists of using DEEDS log data from five different sessions (sessions 2 through 6) along with their corresponding intermediate grades that were attained by all 115 students. Only the first 4 exercises from each session were considered. This has resulted in a total dataset size of 575 entries. We have proceeded in the same direction as in [4] and [31] in terms of data labeling where a student achieving a grade higher than 2 is labeled as class “A” student—(student with “no difficulty”); otherwise, the student is labeled as class “B” student—(student with “difficulty”). By adopting this labeling strategy, 74% of our dataset included students with category A and 26% of students falling under category B.

For the training and evaluation phase, we have considered three sets of experiments. The first experiment consists of a random distribution-based model where we randomly chose 80% of the data (resulting in 460 records ~ 4 data sessions) for training and 20% for testing (115 entries ~ 1 data session). The second experiment is a more generic approach and consists of the classic fivefold cross-validation (resulting in 80% of the data for training and 20% for testing). The third experiment consists of independently assessing the performance of our model per session. In this setup, 4 session data were used for training (equivalent to 80% of the data) and the remaining one (equivalent to 20%) for testing. In the classification phase, we have used five of the well-known classifiers then selected the most accurate one. We have considered MLP, RF, SVM LR, and NB models to classify student performance. Our classifiers’ configuration parameters tuning phase has led to running all classifiers with a *Batch size* equal to 100, *Learning rate* equal to 0.3, and *Loss* equal to 0.1.

We evaluated the effectiveness of the proposed classification model through the analysis of the following four

Table 6 Metrics definitions [26]

Metric	Formula	Description
Accuracy	$\frac{T_p + T_N}{T_p + T_N + F_p + F_N}$	Represents the ratio of the sum of correct classifications with respect to the total number of classifications
Precision	$\frac{T_p}{T_p + F_p}$	Represents the proportion of positive test cases that were classified correctly
Recall (sensitivity)	$\frac{T_p}{T_p + F_N}$	Represents the fraction of actual positive test cases that were properly classified
F1-score	$\frac{2T_p}{2T_p + F_p + F_N}$	Represents the precision and recalls harmonic mean
ROC	$f(\text{TPR}, \text{FPR})$	A probability curve which tells how much a model can distinguish between various classes

Table 7 Model average performance (experiment1—random distribution case)

Classifier	Accuracy	Precision	Recall	F1-Score	ROC
MLP	0.957	0.959	0.957	0.957	0.98
RF	0.974	0.974	0.974	0.974	0.988
SVM	0.948	0.948	0.948	0.946	0.908
LR	92.1	0.934	0.922	0.924	0.918
NB	0.826	0.887	0.826	0.837	0.982

metrics: (1) accuracy, (2) precision, (3) recall (aka. sensitivity), and (4) F1-score. These are briefly described in Table 6, where T_p , T_N , F_p , and F_N represent the True Positive, True Negative, False Positive, and False Negative testing cases, respectively. Along with these four metrics, we also considered the Receiver Operator Characteristic (ROC) metric to analyze the proposed model’s ability to distinguish between classes by looking at the True Positive rate versus the False Positive rates under different settings.

4.2 Model Performance Evaluation

The proposed models were evaluated based on the metrics mentioned in Sect. 4.1. Initially, we have tested our model with the full set of extracted features (86), next we have studied the level of influence of each of the 86 features in the overall model accuracy performance through the application of entropy-based ranking approach.

4.2.1 Model Performance Analysis

Table 7 shows a summary of the obtained results in terms of averages of Accuracy, Precision, Recall, F1-score, and ROC

Fig. 7 Overall model classification performance (Random Distribution of training vs. testing sets)

metrics. These results were collected following our first set of experiments (randomly choosing 80% of the data for training and the remaining 20% for testing), where 5 classifiers were applied (MLP, RF, SVM, LR, and NB). Results show that the RF classifier, which is considered as an ensemble of a combination of tree predictors, has achieved the best performance with 97.4% accuracy and high Recall and Precision values resulting in a high F1-score (97.4%). MLP and SVM (which are known to make good predictions in the case of binary classification problems) have also performed well with an accuracy of 95.7% and 94.8%, respectively, and F1-score of 95.7% and 94.6%, respectively. On the other hand, NB, as expected, did not perform well due to the nature of our dataset wherein features (described in Sect. 3.4) are not completely independent from each other (as a case in point, there is a dependency between activity type feature and the corresponding timing statistics (in terms of duration)). NB has achieved relatively low Accuracy and F1-score (82.6% and 83.7%, respectively).

In line with Table 7, Fig. 7 shows a breakdown of the average performance of all five classifiers in terms of their effectiveness in predicting the correct classes (A—for students with no difficulty, versus B—students with difficulty). Results demonstrated a pattern of improved prediction rates for class A entries versus class B entries across all classifiers. For example, RF achieved 98.8% F1-score for class A entries compared to 95% for class B entries. By contrast, NB consistently showed low performance, especially for class A entries where 73.7% F1-score was attained.

4.2.2 Features Relevance Analysis

In this set of experiment, we have studied the relevance and level of influence of the extracted features through the application of the entropy-based ranking approach. Features ranking results are captured in Table 8. Results show that about 20% of the total features received a low ranking (less than 0.21), indicating that these features may not influence the performance classification model. In fact, a re-run of our prediction model where we have excluded the 18 features from our original dataset has resulted in a very similar

accuracy performance for all classifiers. For instance, for our best performing classifier (RF algorithm), and in the case of random distribution of test data and training data (randomly choosing 80% of the data for training and the remaining 20% for testing), we have achieved an accuracy of 96.7% compared to 97.4% when the full list of features where considered. This insignificant variation in the accuracy performance could be attributed to the size of our dataset. It is worth highlighting that though the achieved accuracy performance was the same, running the model with a reduced number of features contributed to reducing the overall complexity of the model.

Our next results are depicted in Table 9 where we show the corresponding confusion matrix following the execution of MLP, RF, SVM, LR, and NB classifiers. Results are in line with that presented in Fig. 7. For class A entries and when considering the SVM classifier, Table 9 shows that 85 out of 86 were classified properly, which is slightly better than the performance of RF, where 84 out of 86 were classified correctly. However, RF showed a better performance than SVM when classifying class B type of entries (28 out of 29 versus 24 out of 29). It is noteworthy that Table 9 shows that we are in possession of unbalanced data where class A entries outnumber class B entries (86 versus 29, respectively).

Contrary to the results reported in Table 7, where 80% of the data was chosen randomly for training and the remaining 20% for testing, Table 10 shows the performance achieved by all five classifiers while considering fivefold cross-validation, a more conservative approach. Results reported in Table 10 are generally in line with those reported in Table 7 in terms of classifier performances. RF classifier, for instance, performed the best in terms of Accuracy and F1-score (93.37% and 95.4%, respectively), and NB continues to show the lowest performance. It is also noticed that there was a slight decrease of about 4% in the overall performance reported in Table 10 compared to those reported in Table 7, which could be attributed to the random nature of the fivefold cross-validation technique.

Our next set of results aims at studying the performance of our model in a setup where four sessions' data will be used for training, and the remaining unseen session data will be used

Table 8 Ranking of Features using Entropy based Ranking Algorithm

Feature #	Rank	Feature #	Rank	Feature #	Rank	Feature #	Rank
Feat-9	0.509	Feat-6	0.4146	Feat-25	0.3789	Feat-70	0.2263
Feat-31	0.5066	Feat-28	0.4137	Feat-62	0.3785	Feat-45	0.2238
Feat-49	0.497	Feat-3	0.4093	Feat-32	0.3718	Feat-40	0.2039
Feat-85	0.4845	Feat-34	0.406	Feat-74	0.3714	Feat-69	0.2032
Feat-77	0.47	Feat-10	0.4058	Feat-35	0.3683	Feat-43	0.1864
Feat-24	0.4561	Feat-33	0.4041	Feat-2	0.3674	Feat-68	0.1819
Feat-81	0.4439	Feat-22	0.403	Feat-23	0.3636	Feat-67	0.1792
Feat-83	0.4372	Feat-71	0.4013	Feat-19	0.3349	Feat-42	0.1758
Feat-59	0.4371	Feat-8	0.398	Feat-53	0.3293	Feat-64	0.1661
Feat-29	0.4331	Feat-46	0.3979	Feat-20	0.3243	Feat-37	0.1124
Feat-50	0.4331	Feat-26	0.3979	Feat-52	0.3186	Feat-38	0.0865
Feat-1	0.4299	Feat-4	0.3975	Feat-54	0.314	Feat-14	0.0301
Feat-84	0.4283	Feat-78	0.395	Feat-16	0.2918	Feat-12	0.0188
Feat-79	0.4283	Feat-5	0.3947	Feat-18	0.2891	Feat-15	0.0168
Feat-61	0.4282	Feat-60	0.3937	Feat-17	0.2812	Feat-57	0.0139
Feat-47	0.4268	Feat-76	0.3929	Feat-65	0.2803	Feat-11	0
Feat-48	0.4252	Feat-21	0.3922	Feat-66	0.2579	Feat-58	0
Feat-73	0.4239	Feat-7	0.3868	Feat-39	0.2526	Feat-56	0
Feat-27	0.4238	Feat-82	0.3835	Feat-36	0.251	Feat-55	0
Feat-80	0.4176	Feat-72	0.3826	Feat-63	0.2465	Feat-13	0
Feat-86	0.4171	Feat-75	0.3816	Feat-41	0.23		
Feat-51	0.4151	Feat-30	0.3794	Feat-44	0.2275		

Table 9 Class prediction confusion matrices (experiment1—random distribution case)

Classifier	MLP		RF		SVM		LR		NB	
Class	A	B	A	B	A	B	A	B	A	B
A	82	4	84	2	85	1	78	8	67	19
	95%	5%	98%	2%	99%	1%	91%	9%	78%	22%
B	1	28	1	28	5	24	1	28	1	28
	3%	97%	3%	97%	17%	83%	3%	97%	3%	97%

Table 10 Model average performance (experiment 2—fivefold cross-validation)

Classifier	Accuracy	Precision	Recall	F1-Score	ROC
MLP	93.04	0.954	0.948	0.950	0.935
RF	93.37	0.950	0.958	0.954	0.966
SVM	89.90	0.896	0.970	0.932	0.847
LR	89.36	0.943	0.904	0.923	0.898
NB	82.22	0.946	0.796	0.863	0.920

for testing. These are captured in Table 11. Table 11 shows the results of five different experiments. For instance, the set of results for “session ID for testing” equals 2 represents the performance of the model being trained using sessions 1, 3, 4, and 5, then tested on session 2. NB had the poorest performance (between 85 and 88% F1-score), followed by LR, then MLP, then SVM, then RF (between 94 and 97% F1-score).

However, the difference in specific metrics performance varied from one session to another. This was manifested with the Recall and Accuracy metrics where the variations were 20% in session 1 and 14% in sessions 1 and 5, respectively.

4.2.3 Comparative Analysis

The final set of results compares the performance of our proposed model with that of [4, 30, 31], where DEEDS dataset has been used to predict students' academic performance under the same experimental setup. Figure 8 illustrates the performance comparison of all four research results (ours with three others from recent literatures which are named as Hussain, Sriram and Maksud in Fig. 8) in terms of Accuracy, Precision, Recall, and F1-score that were achieved while considering the best performing classifiers (RF in case of our model and ANN in case of the model of Hussain et al. in

Table 11 Per-Session model classification performance (4 data sessions for training and 1 for testing)

Session ID for testing	Classifier	Accuracy	Precision	Recall	F1 score	ROC
1	MLP	0.930	0.951	0.951	0.951	0.928
	RF	0.887	0.897	0.951	0.923	0.964
	SVM	0.887	0.897	0.951	0.923	0.839
	LR	0.904	0.961	0.902	0.931	0.897
	NB	0.800	0.940	0.768	0.846	0.880
2	MLP	0.939	0.952	0.963	0.958	0.955
	RF	0.957	0.943	1.000	0.970	0.988
	SVM	0.922	0.901	1.000	0.948	0.864
	LR	0.913	0.962	0.915	0.938	0.901
	NB	0.852	0.922	0.866	0.893	0.926
3	MLP	0.939	0.987	0.926	0.955	0.948
	RF	0.922	0.962	0.926	0.943	0.961
	SVM	0.904	0.907	0.963	0.934	0.864
	LR	0.904	0.961	0.901	0.930	0.895
	NB	0.817	0.984	0.753	0.853	0.949
4	MLP	0.913	0.973	0.901	0.936	0.917
	RF	0.957	0.975	0.963	0.969	0.964
	SVM	0.887	0.878	0.975	0.924	0.826
	LR	0.878	0.947	0.877	0.910	0.853
	NB	0.843	0.957	0.815	0.880	0.929
5	MLP	0.930	0.940	0.963	0.951	0.934
	RF	0.930	0.940	0.963	0.951	0.945
	SVM	0.895	0.897	0.963	0.929	0.845
	LR	0.895	0.948	0.901	0.924	0.874
	NB	0.807	0.928	0.790	0.853	0.906

Fig. 8 Proposed model performance compared with the results from the recent literature [4, 30, 31]

[4], SVM in case of the models by Siram et al. and Maksud et al. in [30, 31], respectively). Results show that our proposed models outperformed all the three existing models in terms of accuracy with an improvement ranging between 2 to 22% compared to that being achieved in [4, 30, 31]. The F1-score was 12% higher than that achieved in [4] using the ANN classifier and 2 percent in the case of the SVM classifier being used in [30, 31]. We believe that such improvement

is attributed to the extended set of features that were introduced by our model compared to a reduced and abstract list of five features per exercise proposed in [4]. While in [4], the authors did not differentiate between the types of activities within a single exercise, our model provisioned for the different types of activities resulting in 9 distinct features along with the total activity occurrence count per exercise. This has resulted in a total of 10 features related to activities per exer-

cise. Also, contrary to the model in [4], where the interaction of students with DEEDS was only captured by a single feature counting the number of keystrokes, our model took into account the different types of interactions of students with DEEDS using input peripherals (mouse and keyboard) leading to a total of 6 features per exercise. The extra information that was provided to the prediction model explains the significant classification performance improvement captured in Fig. 8.

5 Conclusion

In this article, we demonstrated the ability to predict student performance by analyzing the interaction logs of students in the DEEDS dataset. We have extracted a total of 86 statistical features, which are categorized into three main categories based on different criteria: (1) activity-type based, (2) timing statistics-based, and (3) peripheral activity count-based features. This set of features was further reduced during the feature selection phase where we have applied the entropy-based selection technique and only influential features were retained for training purposes. We trained our model considering three different scenarios: (1) 80:20 random data split for training and testing, fivefold cross-validation, and (3) train the model on all sessions but one which will be used for testing. Then we collected performance results in terms of Accuracy, Precision, Recall, F1-score, and ROC, using the five prominent classifiers (RF, SVM, MLP, LR, and NB).

Results showed that the best performance was obtained using an RF classifier with a classification accuracy of 97% and an F1-score of 97%. However, the poorest results were achieved with NB due to the inherent dependency of the model on the proposed features. When comparing our model with the benchmark models proposed by Hussain et al. in [4], Sriram et al. in [30] and Maksud et al. in [31], we were able to demonstrate that, under a similar experimental setup, our model outperformed existing models in terms of classification accuracy and F1-score.

Future work For future work, we propose exploring various research directions as follows:

1. Modify and compare the proposed model with a model which considers more sophisticated Machine Learning algorithms for feature extraction and classification such as Decision Trees, fuzzy entropy-based analysis, and transfer learning, etc.
2. Enhance the prediction models to a multi-label problem aimed at classifying students into four broad categories: (1) very weak, (2) weak, (3) average, and (4) good.
3. Consider proposing a regression model to predict exam grades along with classifying students' performance using just the binary classification approach.

Funding Not applicable.

Availability of Data and Materials The dataset used in this study is publicly published <https://sites.google.com/site/learninganalyticsforall/data-sets/epm-dataset>.

Declarations

Conflict of interest The author has no conflicts of interest.

References

1. Vahdat, M.; Oneto, L.; Anguita, D.; Funk, M.; Rauterberg, M.: A learning analytics approach to correlate the academic achievements of students with interaction data from an educational simulator. In: *Design for Teaching and Learning in a Networked World*, pp. 352–366. Springer, Cham (2015)
2. Tomasevic, N.; Gvozdenovic, N.; Vranes, S.: An overview and comparison of supervised data mining techniques for student exam performance prediction. *Comput. Educ.* **143**, 103676 (2020)
3. Hellas, A.; Ithantola, P.; Petersen, A.; Ajanovski, V.; Gutica, M.; Hynninen, T.; Liao, S.N.: Predicting academic performance: a systematic literature review. In: *Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*, pp. 175–199 (2018)
4. Hussain, M.; Zhu, W.; Zhang, W.; Abidi, S.M.R.; Ali, S.: Using machine learning to predict student difficulties from learning session data. *Artif. Intell. Rev.* **52**(1), 381–407 (2019)
5. Buenaño-Fernández, D.; Gil, D.; Luján-Mora, S.: Application of machine learning in predicting performance for computer engineering students: a case study. *Sustainability* **11**(10), 2833 (2019)
6. Ofori, F.; Maina, E.; Gitonga, R.: Using machine learning algorithms to predict students performance and improve learning outcome: a literature based review. *J. Inf. Technol.* **4**(1), 33–55 (2020)
7. Huang, S.; Fang, N.: Predicting student academic performance in an engineering dynamics course: a comparison of four types of predictive mathematical models. *Comput. Educ.* **61**, 133–145 (2013)
8. Rastrollo-Guerrero, J.L.; Gomez-Pulido, J.A.; Duran-Dominguez, A.: Analyzing and predicting students' performance by means of machine learning: a review. *Appl. Sci.* **10**(3), 1042 (2020)
9. Sundar, P.P.: A comparative study for predicting students academic performance using Bayesian network classifiers. *IOSR J. Eng. IOSR-JEN e-ISSN, 2250-3021* (2013)
10. Burgos, C.; Campanario, M.L.; de la Peña, D.; Lara, J.A.; Lizcano, D.; Martínez, M.A.: Data mining for modeling students' performance: a tutoring action plan to prevent academic dropout. *Comput. Electr. Eng.* **66**, 541–556 (2018)
11. Ma, X.; Zhou, Z.: Student pass rates prediction using optimized support vector machine and decision tree. In: *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 209–215. IEEE (2018)
12. Masci, C.; Johnes, G.; Agasisti, T.: Student and school performance across countries: a machine learning approach. *Eur. J. Oper. Res.* **269**(3), 1072–1085 (2018)
13. Pardo, A.; Han, F.; Ellis, R.A.: Combining university student self-regulated learning indicators and engagement with online learning events to predict academic performance. *IEEE Trans. Learn. Technol.* **10**(1), 82–92 (2016)
14. Gray, G.; McGuinness, C.; Owende, P.: An application of classification models to predict learner progression in tertiary education. In:

- 2014 IEEE International Advance Computing Conference (IACC), pp. 549–554. IEEE (2014)
15. Hussain, M.; Zhu, W.; Zhang, W.; Abidi, S.M.R.: Student engagement predictions in an e-learning system and their impact on student course assessment scores. *Comput. Intell. Neurosci.* (2018)
 16. Elbadrawy, A.; Studham, R.S.; Karypis, G.: Collaborative multi-regression models for predicting students' performance in course activities. In: *Proceedings of the Fifth International Conference on Learning Analytics and Knowledge*, pp. 103–107 (2015)
 17. Liu, S.; d'Aquin, M.: Unsupervised learning for understanding student achievement in a distance learning setting. In: *2017 IEEE Global Engineering Education Conference (EDUCON)*, pp. 1373–1377. IEEE (2017)
 18. Kuzilek, J.; Hlosta, M.; Herrmannova, D.; Zdrahal, Z.; Vclavek, J.; Wolff, A.: OU Analyse: analysing at-risk students at The Open University. *Learn. Analyt. Rev.* 1–16 (2015)
 19. Ho, T.K.: Random decision forests. In: *Proceedings of 3rd International Conference on Document Analysis and Recognition*, vol. 1, pp. 278–282. IEEE (1995)
 20. Bauer, E.; Kohavi, R.: An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Mach. Learn.* **36**(1), 105–139 (1999)
 21. Latif, G.; Iskandar, D.A.; Alghazo, J.M.; Mohammad, N.: Enhanced MR image classification using hybrid statistical and wavelets features. *IEEE Access* **7**, 9634–9644 (2018)
 22. Suthaharan, S.: Machine learning models and algorithms for big data classification. *Integr. Ser. Inf. Syst* **36**, 1–12 (2016)
 23. Misra, S.; Li, H.; He, J.: *Machine Learning for subsurface Characterization*. Gulf Professional Publishing, Oxford (2019)
 24. Bewick, V.; Cheek, L.; Ball, J.: Statistics review 14: logistic regression. *Crit. Care* **9**(1), 1–7 (2005)
 25. Meurer, W.J.; Tolles, J.: Logistic regression diagnostics: understanding how well a model predicts outcomes. *JAMA* **317**(10), 1068–1069 (2017)
 26. Rehman, A.; Naz, S.; Razzak, M.I.; Hameed, I.A.: Automatic visual features for writer identification: a deep learning approach. *IEEE Access* **7**, 17149–17157 (2019)
 27. Trstenjak, B.; Đonko, D.: Determining the impact of demographic features in predicting student success in Croatia. In: *2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp. 1222–1227. IEEE (2014)
 28. Kursa, M.B.; Rudnicki, W.R.: Feature selection with the Boruta package. *J Stat Softw* **36**(11), 1–13 (2010)
 29. Shaw, R.G.; Mitchell-Olds, T.: ANOVA for unbalanced data: an overview. *Ecology* **74**(6), 1638–1645 (1993)
 30. Sriram, K.; Chakravarthy, T.; Anastraj, K.: A comparative analysis of student performance prediction using machine learning techniques with DEEDS lab. *J. Compos. Theory* **XII**(VIII) (2019)
 31. Maksud, M.; Nesar, A.: Machine learning approaches to digital learning performance analysis. *Int. J. Comput. Digit. Syst.* **10**, 2–9 (2020)
 32. Leena, H. A.; Ranim, S. A.; Mona, S. A.; Dana, K. A.; Irfan, U. K.; Nida, A.: Predicting Student Academic Performance using Support Vector Machine and Random Forest. *3rd International Conference on Education Technology Management*. pp. 100–107 (2020)
 33. Hasan, R.; Sellappan, P.; Salman, M.; Ali, A.; Kamal, U.S.; Mian, U.S.: Predicting student performance in higher educational institutions using video learning analytics and data mining techniques. *Appl. Sci.* **10**(11), 3894 (2020)
 34. Aydoğdu, Ş.: Predicting student final performance using artificial neural networks in online learning environments. *Educ. Inf. Technol.* **25**(3), 1913–1927 (2020)
 35. Biesiada, J.; Włodzisław D.; Adam K.; Krystian M.; Sebastian P.: Feature ranking methods based on information entropy with parzen windows. In: *International Conference on Research in Electrotechnology and Applied Informatics*, vol. 1, p. 1. (2005)
 36. Horino, H.; Hirofumi, N.; Elisa, C.A.C.; Toru, H.: Development of an entropy-based feature selection method and analysis of online reviews on real estate. In: *IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, pp. 2351–2355. IEEE (2017)