IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 16, 2020, accepted October 29, 2020, date of publication November 6, 2020, date of current version November 19, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3036572

Predicting Student Performance and Its Influential Factors Using Hybrid Regression and Multi-Label Classification

ABDULLAH ALSHANQITI AND ABDALLAH NAMOUN
Faculty of Computer and Information Systems, Islamic University of Madinah, Madinah 42351, Saudi Arabia

Corresponding author: Abdullah Alshanqiti (a.m.alshanqiti@gmail.com; a.namoun@iu.edu.sa)

This work was supported by the Deanship of Scientific Research, Distinguished Project, Islamic University in Madinah, under Grant #22 − 2018 − 2019.

ABSTRACT Understanding, modeling, and predicting student performance in higher education poses significant challenges concerning the design of accurate and robust diagnostic models. While numerous studies attempted to develop intelligent classifiers for anticipating student achievement, they overlooked the importance of identifying the key factors that lead to the achieved performance. Such identification is essential to empower program leaders to recognize the strengths and weaknesses of their academic programs, and thereby take the necessary corrective interventions to ameliorate student achievements. To this end, our paper contributes, firstly, a hybrid regression model that optimizes the prediction accuracy of student academic performance, measured as future grades in different courses, and, secondly, an optimized multi-label classifier that predicts the qualitative values for the influence of various factors associated with the obtained student performance. The prediction of student performance is produced by combining three dynamically weighted techniques, namely collaborative filtering, fuzzy set rules, and Lasso linear regression. However, the multi-label prediction of the influential factors is generated using an optimized self-organizing map. We empirically investigate and demonstrate the effectiveness of our entire approach on seven publicly available and varying datasets. The experimental results show considerable improvements compared to single baseline models (e.g. linear regression, matrix factorization), demonstrating the practicality of the proposed approach in pinpointing multiple factors impacting student performance. As future works, this research emphasizes the need to predict the student attainment of learning outcomes.

INDEX TERMS Student performance, influential factors, hybrid approach, collaborative filtering, matrix factorization, fuzzy set theory, Lasso linear regression, self organizing map, neural networks, multi-label classification, machine learning.

I. INTRODUCTION

Despite the recent paradigm shift in higher education (e.g., outcome-based education [1], [2]), many colleges and universities worldwide still suffer from poor student performance [3]. For example, only around 40% of first-time college students enrolled in a 4-year bachelor's degree graduate within four years in the US [4], while the college dropout rate reached a whopping 40% in 2018 [5]. To overcome the repercussions of this phenomenon, various research studies attempted to develop automatic models that predict future student academic performance in higher education [6]–[8]. The timely prediction of student performance offers a myriad of benefits, including the early identification of students struggling to pass their modules and those at risk of dropping out, course selection pathways, as well as the attributes that influence student retention rates and behaviors. Such intelligent insights empower educational leaders to devise and implement corrective interventions to support academic advising, guide curriculum changes and improvements, and determine the pitfalls of the programs [9]. However, selecting an appropriate machine learning model to estimate student performance accurately remains a complex endeavor [10].

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

Findings have shown that student academic achievements are typically influenced by a myriad of factors, ranging from academic and non-academic attributes [10]. The variability of these factors necessitates the development of a complex predictive model. Ensemble (i.e., hybrid) learning models are already proven to outperform individual learning models concerning the prediction accuracy of student academic performance [11], [12]. Moreover, a recent survey revealed that around 50% of the studies employ supervised learning algorithms, while only 5% of the studies use unsupervised approaches [8]. Indeed, supervised machine learning provides acceptable prediction accuracy [10]; however, we believe that augmenting a supervised model with unsupervised learning will produce even more accurate predictions, with fewer generalization errors. As such, our suggested model integrates the powers of supervised and unsupervised learning.

The available models and algorithms focus mainly on improving the prediction accuracy of future student performance [8], [13]. They fall short of generating an explanatory analysis of the exact factors (i.e., variables) that cause the observed student performance. Additionally, relying on a single model, whether this model is linear or non-linear, may be insufficient due to the difficulty of capturing a variety of factors in one predictor model. Factors affecting student performance often differ significantly among students and between academic semesters for the same students. To avoid ambiguity, single linear models would generally suffer from underfitting the data on which they were trained (i.e., consisting of many overlapping student behaviors), leading to high rates of false predictions. Likewise, false predictions would also be high with non-linear models as they are prone to overfitting just parts of the data on which they trained (i.e., the model would only remember some aspects of student behaviors).

Moreover, existing ensemble machine learning solutions do not accommodate for a dynamic weighted contribution of the participating models in predicting student performance. Further limitations concern the disuse of a training set or the use of a single dataset to validate the model, such as [13]. Moreover, some models focus on predicting the achievements of first-year students only (e.g., [14]). More than 50% of the surveyed studies used SVM and ANN techniques to predict student performance [8]. Moreover, most related approaches that we are aware of are confined to predicting future course grades only without associating them with the key factors that lead to the obtained student performance [8], [13]. In our view, understanding the impact of those enabling and inhibiting factors is quite essential to devise corrective plans to improve student achievements and reduce the risk of dropout. Our approach is distinguished by its clustering feature that helps in understanding the associated factors leading to the predicted future course grades.

To address the aforementioned limitations, we contribute a hybrid regression approach along with a semi-supervised learning technique for identifying the enabling factors and inhibitors of student performance in educational programs. The proposed approach seeks to optimize the prediction accuracy of student academic performance based on course grades, and then identify the possible factors that might have caused the observed student achievements. We assume that program strengths and weaknesses are instigated by a set of factors and circumstances, which are believed to have direct or latent effects on student academic results. More precisely, the main contributions offered by this research include:

- Combining the powers of three distinct techniques, namely (1) collaborative filtering technique (i.e., implemented based on matrix factorization), (2) fuzzy rules, and (3) Lasso linear regression, to obtain robust predictions of future student performance. The novelty here lies in (1) the inference of approximately 106 fuzzy rules for estimating students' grades in each course, and more importantly, in (2) the integration of these three distinct techniques through the use of our proposed weighted sum model. This model allows one to determine and adjust the importance weights for the three techniques dynamically according to each student circumstances.
- Optimizing the use of Self Organizing Map (i.e., which is an unsupervised learning method through neural networks) as a multi-label classifier by introducing a weighted mean scheme. The multi-label self-organizing map anticipates multiple factors that might lead to a particular student academic performance. Here, one of the advantages of our proposed weighted mean scheme is that it can significantly reduce the sensitivity of the self-organizing map model to the neighborhood radius when treating it as a multi-label classifier.
- Implementing a proof of concept of the entire approach, which can be applied to similar educational programs with minimal customization.

We conducted extensive performance evaluation (i.e., measured using root mean square error - RSME) of our prototype against primary student performance prediction techniques using seven representatives benchmark datasets. Our empirical results demonstrate the performance efficiency and practical benefits of our predictive models (i.e., the hybrid and the multi-label classifier).

The remainder of this paper is divided into six sections. The next two sections (II and III) review the related works by highlighting the research gaps and shedding light on the factors deemed to influence student performance and learning outcomes. Section IV details the steps of our approach, while section V presents the analysis and results of our evaluation. Section VI discusses the results and draws attention to the shortcomings of the proposed approach. Lastly, section VII concludes the key findings and suggests two promising future research directions.

II. REVIEW OF RELATED WORKS

Since our research focuses on computer-supported predictive analytics, we reviewed the related works from two main perspectives. The first part introduces the foundational concepts of student academic performance, and the second part explores the modern approaches used for predicting and explaining student performance.

A. STUDENT ACADEMIC PERFORMANCE

Typically, data analytics of educational data involve two main strands, namely predictive analytics and learning analytics [15]. Predictive analytics seeks to predict student learning and performance, identify student failure rates, and recommend future courses that yield the best results [16]. However, learning analytics seeks to collect and analyze student learning data and their environment to improve the attainment of student learning outcomes [17]. Moreover, data mining techniques have been applied to student data to unravel the link between factors and student learning [18]. For example, ethnicity was found not to impact students' cumulative grade point scores [19]. Similarly, [20] revealed that cognitive university admission requirements did not accurately explain student performance, suggesting that non-academic factors may play a vital role in student learning. Our research attempts to improve the prediction of student performance and explain the obtained predictions, thus applying both predictive and explanatory modeling of student performance.

Previous works define student performance as a measure of student proficiency and achievements in the upcoming courses [21]. Indeed, assessing student academic performance has been for long a significant goal in higher education to overcome persisting issues, including low academic grades, increased student failure and dropouts, and prolonged graduation time, among others [13]. The predictions of student achievements are calculated mainly using previous semesters' grades and current coursework assessments, such as assignments, midterms, and projects, and final exams [13]. However, subsequent works explored the influence of non-academic traits, such as student demographics and socioeconomic status, on student achievements [22]–[24]. Despite their significance, the assessment of learning outcomes has been used less frequently to measure student performance [13], [25]. Past findings indicate that several intertwined factors impact student performance, and their precise prediction requires the development of sophisticated models.

The prediction of student performance in higher education is a worthwhile activity for it accomplishes strategic benefits, such as the development of early warning and course path recommendation systems, the detection of adverse student behaviors, and the automation of course assessment [13], [16], [26]. However, the accurate prediction of student academic achievements is a complicated research endeavor, requiring an in-depth understanding of all aspects and circumstances surrounding the students and their learning environment [9]. Moreover, predicting student performance involves discovering student behaviors and preferences and considering various influential academic and non-academic factors [15]. However, the current findings are still unsatisfying given that (1) the prediction accuracy of single learning models remains low, e.g., [27], [28], and (2) the factors leading to the observed academic performance are overlooked or diagnosed inadequately, e.g., [29], [30]. Our research aims to bridge these major gaps.

Some research works, e.g. [31], went beyond predicting course grades to identifying at-risk students. However, accurate predictive modeling in education remains challenging due to data sparsity and exponentiality problems with powerful classification models, such as the Support Vector Machine (SVM) [9]. To tackle, as an example, the latter challenge for the SVM, [32] applied a multivariate normal approach and vector transformations to reduce the training time of the model. Although the training time was reduced by approximately 59%, the optimized algorithm still achieved a promising accuracy of 93% in recognizing the most vulnerable students for failure. In subsequent work, [33] developed a generative adversarial network-based deep support vector machine (denoted as ICGAN-DSVM) model, which handles small training datasets and produces high accuracy predictions of student performance. The results showed that family tutoring, combined with school tutoring, improves student performance. Although combining existing approaches (e.g., CGAN) improved the predictions by up to 29%, small validation datasets were used to verify the model performance. Other researchers showed that learning discriminant analysis and support vector machine yielded the best classification results of project grades when training small datasets, as in the case of postgraduate programs [29]. However, these works focus primarily on predicting students who might not pass future courses and do not provide a comprehensive explanation of the factors leading to their failure.

In [34], the authors suggested a multi-view approach that uses genetic programming classification rules to identify the underperforming students who particularly suffer from socio-economic disadvantages. This approach combines several sources of student data to solidify the feedback recommended to decision-makers. However, their proposed architecture does not identify the factors leading to the predicted performance. In a similar fashion, [35] proposed a genetic algorithm, as part of an early warning system, to detect early students' dropout from courses. Again, this warning system does not justify the motives behind the possible dropout of students.

B. APPROACHES TO PREDICTING STUDENT PERFORMANCE

The abundance of educational data [36] coupled with the emergence of predictive modelling [9], [37] empower the creation of effective learning analytics models that inform educational institutions about the future academic performance of students to assist them enhance the learning processes.

Learning analytics explore student data to investigate their activities and behaviors and provide relevant recommendations [38]. High-level categories of learning analytics models range from statistical techniques, educational data mining methods to advanced machine learning models [6], [7], [13]. Specific examples of prominent algorithms predicting future academic achievement of students include regression models [39], decision trees [17], collaborative filtering [40], support vector machine [29], and artificial neural networks [8]. However, most of these techniques have been used separately (i.e., as single models), focusing mainly on supervised learning to predict student performance [13], [29]. In this research, we endeavor to combine supervised and unsupervised learning to anticipate student performance more accurately.

When it comes to predicting student performance, previous studies showed that supervised learning techniques have been the more popular and preferred choice [10]. For instance, [8] revealed that the SVM is the most used method for predicting student academic performance, while the artificial neural network was the least favored technique. However, [13] showed that statistical linear modeling techniques are used more than other methods to calculate and predict student success. For example, [20] demonstrated that linear and quadratic regression models could be used to predict the final cumulative grade point average to an acceptable level based on the grade point average of the first three years. Reference [17] developed a decision tree model of entropy and information gain values of formative assessments to predict the risk of failing in the summative assessments. The tree helps course instructors identify students who require learning support to pass the final exam. In contrast to those proposals, fewer studies have attempted to use unsupervised learning to predict and explain student performance [8], [13], [26].

Matrix factorization approaches have been explored recently by decomposing the student-course matrix into two low-dimensional matrices (e.g., a student matrix and a course matrix) such that student performance can be estimated by calculating the product of these two low-dimensional matrices [42]. Such matrices can explain the variability of student grades [44]. References [27], [44] suggested that analyzing course-specific data yielded more accurate grade predictions than traditional approaches, such as regression models and student-based collaborative filtering. In their approach, low-rank matrix factorization methods and linear regression models were applied to historical course grades. However, grade predictions were less plausible when a student-course-specific approach was applied. Other works incorporated additive latent effect models (i.e., ALE) within matrix factorization methods to calculate future course grades for the next terms [41]. The proposed ALE models incorporate four factors related to instructors, student academic level, student interest, and knowledge to predict future grades. Unlike the works of [27], [44], our work extends collaborative filtering with a regression model and a fuzzy rule-based model to predict student grades and explain the predictions through multiple factors, using a self-organizing map. This is in line with the recommendation that ensemble classifiers produce more accurate predictions [45].

There is substantiated evidence in recent literature claiming that combining multiple machine learning classifiers would improve the prediction results, with prediction accuracy improvements ranging from 25% to 30% [16], [46], [47]. For instance, [48] integrated several classification algorithms to predict student performance through a voting mechanism. The hybrid classifier achieved an accuracy percentage of 92.59%, an increase of at least 4% compared to other individual approaches such as decision trees, K-nearest neighbor, and multilayer perception. Similarly, [49] showed that a hybrid model that combines support vector machines and K-means clustering could predict the number of attempts before passing a course and course grades based on past student performance. Linear discriminant analysis was applied subsequently to specify the most relevant factors that yield particular student achievements. Moreover, ensembles of the decision tree, support vector machine, and artificial neural network were applied on multiple student data sources to identify students at risk and predict student grades [47]. Results showed that a stacked hybrid model is more efficient and accurate (about 81%) and introduces fewer prediction errors than single classifiers (i.e., SVM, ANN, and decision trees). Moreover, the rotation forest ensemble gave the highest prediction accuracy of student performance (i.e., 76%) [11]. Meanwhile, the RMSE ranged from 0.41 to 0.44. Although these hybrid models fared better than the single models, the prediction accuracy levels are still low, while the prediction error remains relatively high. Furthermore, the existing hybrid models focus mainly on predicting future student grades or dropout rates without making inferences to the causes. Therefore, it is imperative to explore other combinations of predictive methods that anticipate student performance accurately and highlight the enabling factors of a successful academic program. Our work is one step towards this vision. Table 1 highlights the most prominent models predicting student performance, along with their strengths and weaknesses.

C. THE RESEARCH GAPS

Predicting student achievements accurately is a complex task, necessitating new intelligent approaches that consider the evolving factors and circumstances, which influence student academic performance. The impact of these factors and circumstances may differ from one batch of students to another and from one program to another. Our extensive review of the related works revealed gaps pertaining to the following areas:

- Lack of hybrid approaches, which combine the advantages of supervised and unsupervised learning to automate and optimize the prediction accuracy of student academic performance.
- The inflexibility of existing models to analyze multiple academic and non-academic factors that are deemed to influence the quality of student learning.

TABLE 1. A Summary of Existing Works (i.e., [27]–[31], [41]–[43]) Predicting Student Performance.

Method	Focus of Study	Evaluation	Source of datasets	Observation (+ strength and - weak points)
Linear regression model and matrix factorization [27]	Predicting student grades for future courses for personalized degree paths	RMSE = [0.63, 0.72] Precision is 26.68%	USA, University of Minnesota **Private** dataset, consisting of (2949 undergraduate students, 2556 different courses, 76,748 student–course grades, and 2 Majors)	+ The study focused on grade letter prediction. + A course-specific subset of data resulted in the best predictions. - Performance differed significantly between different departments and depended on prior courses. - Only grade prediction, no prediction of student marks. - The student course-specific approach gave low predictions.

Attention graph convolutional network model. [31]	Predicting students' next term course grades and identifying students at risk of failing or dropping out.	MAE = [0.30, 0.54]
		Precision = [80.21%, 93.23%]
		Detection of at-risk students [43.8%, 68.5%]
			USA, George Mason University **Private** dataset, consisting of (43490 undergraduate students, 185 courses, 385505 grades, 5 Majors)	+ Grade letter prediction.
				+ Tests were performed on two different terms.
				+ The model provides explanations about the predictions in the form of previous courses.
				+ Dependency between courses was taken into consideration (i.e., student knowledge evolution).
				- Model performance varies across the majors.
				- The explanation is linked to previous courses only; other attributes are ignored.
				- MAE was relatively high for some majors.

Five models explored: ANN, SVM, KNN Learning Discriminant Analysis (LDA) Naïve Bayes (NB) [29]	Predicting student performance at the postgraduate level using small datasets.	Precision = [58.1%, 69.7%]	Emirates, British University in Dubai. **Private** dataset, consisting of (50 Postgraduate Students, 9 courses, 311 instances, 1 Major)	+ The approach was able to train and model small datasets that are appropriate for postgraduate studies. + Key predictors in small datasets were discovered. - Best performance indicators were extracted from a heat map, which might be inaccurate. - Only 5 student attributes were used to predict the performance; other variables were ignored. - Only 4 encoding grades (labels) were used.

Bayesian deep learning approaches (MLP and LSTM) [30]	Predicting student grades. Estimating the uncertainty associated with the performance predictions. for identifying the influential courses for student success.	MAE = [0.253, 0.588]
		Precision = [79.32%, 92.62%]
			USA, George Mason University. **Private** dataset, consisting of (28717 undergraduate students, 182 course, 249716 grades, 5 Majors)	+ The model considers courses of previous semesters; thus, the knowledge accumulated by students.
				+ The model suggests at-risk students and explains the performance predictions in the form of prior influential courses leading to student failures.
				- Significant variance in performance prediction between majors.
				- No statistical testing was performed to assert the superiority of the proposed models.

Matrix factorization [41]	Predicting next term grades based on several latent factors such as student academic level and course instructors.	MAE= [0.615, 0.654]
		Precision= [63.8%, 67.0%]
			USA, George Mason Uni. **Private** dataset, consisting of (11027 undergraduate students, 1318 courses, 140259 grades, 8 Majors)	+ The approach combines matrix factorization with additive latent factors.
				+ Experiments were performed over several majors.
				+ Factors influencing course selection were demonstrated.
				- The model works well on specific majors only.
				- The model covered 4 latent factors only

Discriminative and generative classification models: SVM, C4.5, CART Bayes Network, and Naïve Bayes [42]	Predicting student completion of degrees using personalized features, such as family expenditures.	Precision = [71%, 86.7%]	Pakistan, various universities **Private** dataset, consisting of 776 student records.	+ The study explored 23 features; for example, family expenditures and student personal data are used to predicting student success. + The model combines features to predict student performance. - Student grades were not predicted, only success or failure. - Not all personal features were considered in the prediction of student success. - The model testing was done on a small dataset.

Multiple Linear Regression (MLR) [28]	Predicting students' final scores using multivariate regression.	pMSE = 198.62
		pMAPC= 0.81
			Taiwan, National Central University **Private** dataset, consisting of 58 undergraduate students; 1 course; 1 major)	+ PCA was proved to improve the performance predictive accuracy once added to the MLR.
				+ The use of six components was determined to achieve the best predictions.
				- The model was tested on one blended course only; thus, it does not apply to other courses and majors.
				- It is not possible to generalize the results due to the small dataset.

Below single and hybrid approach: decision tree, Logistic Regression , SVM, MLP, NBC, KNN. Hybrid (NBC + SVM + KNN) [43]	Identifying at-risk students using standards-based grading collected during the academic term.	Precision = [34.5%, 86.2%]	USA, San Jose State University **Private** dataset, consisting of (2973 undergraduate students; 1 course, 1 major)	+ Use of a hybrid model encompassing three best predictor methods (NBC + SVM + KNN). + Use of feature selection to reduce the number of predictive variables. + The focus was on the achievements of students concerning the learning objectives of the course. - Use of semester data to predict the end of term student success. - Not possible to generalize since it was tested on a single course. - Need to build prediction models for specific courses. - Feature selection did improve the predictions but not significantly.

MLP: Multilayer Perceptron
SVM: Support Vector Machine
pMSE: Predictive Mean Square Error
pMAPC: Predictive Absolute Percentage Correction
ANN: Artificial Neural Network
LSTM: Long Short-Term Memory
RMSE: Root Mean Square Error
MAE: Mean Absolute Error
CART: Classification and Regression Tree
KNN: K-Nearest Neighbor
NBC: Naïve Bayes Classifier

TABLE 2. A Review of Related Studies (i.e., [11], [27], [41], [43], [45], [50], [51]) and their Research Gaps (denoted as RG).

	RG1	RG2	RG3	RG4
[50]	• Single unsupervised learning model: use of only the self-organizing map for multi-label classification. • No prediction of student performance.	• Not applied to the context of student performance. • Only the winning neuron is used to classify new instances.	No	Yes • But the approach was tested on seven non-academic datasets.
[51]	• Single unsupervised learning model: use of only the self-organizing map for multi-label classification. • No prediction of student performance	• Not applied to the context of student performance. • Only nearby neurons are used to classify new instances.	No	Yes • But the approach was tested on seven non-academic datasets.
[27]	• Single supervised learning models: linear regression, matrix factorization. • Prediction of letter grades.	• No identification of influential factors.	No	Yes • The approach was tested on 2 datasets only.
[41]	• Single supervised learning model: matrix factorization. • Prediction of next term grades.	Yes • However, the model was restricted to only four predetermined factors.	No	Yes • The approach was tested on one academic dataset involving 8 majors.
[11]	• Ensemble supervised learning model: bagging, boosting, random forest, rotation forest. • Prediction of pass / fail only. Prediction focuses on the first year of degree only.	No	No	No • Only one dataset was used.
[43]	• Ensemble supervised learning model composed of support vector machine, K-nearest neighbours, and naïve bayes. And feature selection method (correlation). • Prediction of at-risk students.	Yes • However, only course assessments were considered. • In-semester assessment data are used.	No	No • Only one dataset for a single course was used to verify the hybrid model.
[45]	• Ensemble semi-supervised learning model containing naives bayes, the multilayer perceptron, the sequential minimal optimization, the logistic model tree, the PART, and 3-nearest neighbour. • Prediction of grade classification in the final exams.	Yes • However, only course assessment attributes were considered in the prediction.	No • A simple majority voting is used; thus, ignoring some classifiers.	No • The model was tested on one course only.
Our	Yes • Weighted Hybrid Regression and Multi-Label Self Organizing Map. • Combines supervised and unsupervised approaches. • Prediction of student grades/marks.	Yes • Prediction of performance and all associated factors influencing the performance. • A weighted scheme is used to consider all neurons.	Yes • A weighted scheme is applied to dynamically change the contribution of each participating model in the prediction of student performance	• 7 datasets, including academic and non-academic datasets.

RG1: A hybrid model is used to improve the accuracy of performance predictions
RG2: Enabling and inhibiting factors of student achievements are included in the prediction
RG3: Dynamic adjustment of participating models in the prediction
RG4: Model is validated using multiple datasets
Yes= RG addressed, No= RG not addressed

Some approaches predict student achievements without associating them with the enabling factors or possible weaknesses, while others consider only a small subset of potential factors.

- Models composing hybrid approaches do not adjust their contribution dynamically in estimating the predictions according to student circumstances.
- Many student prediction models are validated using a single dataset, which is considered as a threat to the validity of the model.

Table 2 summarizes the most related works predicting student achievements and indicates their weaknesses by linking them to four research gaps that we attempt to address in this research.

III. FACTORS IMPACTING STUDENT ACADEMIC PERFORMANCE

Before delving into the factors and circumstances that might drive student performance, we first need to define the concept of academic success. Although many researchers refer to academic success merely by academic achievement in different courses, broader views emphasise a multifaceted interaction of components including the successful engagement and completion academic activities, as well as the attainment of the intended learning outcomes by students to prepare them for the job market [52]. [53] highlights six main measures to determine academic success, namely academic achievement (e.g. in the form of course grades and GPA), attainment level of learning outcomes (e.g. course evaluation), perceived satisfaction (e.g. surveys), acquisition of skills and competencies, career success, and persistence (e.g. graduation rates).Reference [52] reports that previous academic achievements of students and their demographics were the most important factors that can be used to predict academic success in university settings. Previous academic achievement refers to student grades obtained both pre-university (e.g. high school) and during university. Student demographics resulted in performance differences, particularly with respect to gender, age and ethnicity. Other influential factors include student psychological traits, learning environment, and e-learning activity.

The accurate prediction of student academic performance necessitates a deep understanding of the factors and features that impact student results and the attainment of student outcomes. To this end, [13] surveyed 357 relevant papers in the area of student performance detailing the impact of 29 features. These features were mainly related to course and pre-course performance, student engagement, student demographics such as gender, high school performance, and psychomotor skills, such as self-regulation. However, the degree dropout rate was mainly influenced by student motivation, habits, social and financial issues, lack of progress, and career transition.

Another important survey by [8], exploring 71 papers, indicated that 70% of the studies aimed at predicting student performance and 21% of the studies targeted the prediction of student dropout probability. The predications are calculated based on past student grades, demographics and characteristics. Again, this survey shows lack of efforts to predict the enabling and inhibiting factors against student success. Moreover, most of the previous research attempts used a small number of factors to predict student performance. It is worthy of note that summative performance metrics, such as the GPA score or range, have been used less frequently (less than 13% of the studies) in the prediction of student performance despite its importance [13].

In this research, we go beyond the mere goal of modelling and predicting student performance using student-course features to examining and determining the key factors leading to the obtained student performance at the program level using unsupervised learning. Hence, one of our aims is to predict qualitative values for the influential factors after predicting student performance. In a broader sense, capturing a deeper understanding of the factors might assist us in revealing the key strengths and weaknesses behind the attainment of the student learning outcomes.

IV. METHODOLOGY

The hybridization of intelligent models is conceptually vital for addressing complex real-world problems, where the participating models can combine their relative strengths while overcoming any potential weaknesses. In accordance with its significant advantages, we propose a hybrid predictive model that combines a fuzzy rule-based technique along with two machine learning techniques, as illustrated in Figure 1. Initially, given a specific dataset that describes student-course features, the hybrid regression model attempts to predict future student performance. This is followed by the application of an unsupervised neural network model to predict the main qualitative factors that justify the obtained student achievements. In the follow-up stage, rolling up such predicted information (i.e., from student level to program level) would intelligently assist us in providing and justifying insights into the main reasons behind the attained outcomes for an educational program.

We begin by formalizing our research problem and then delve into the proposed solution. At the top part of Figure 1(A), the dataset structure is abstracted and described in terms of the requisite type of features and labelled classes. Such dataset might be extracted from an educational information system that stores student records in chronological sequence, typically by annually academic semesters. In this approach, we assume splitting dataset into two parts based on time-series such that student records from the past and the currently running semesters are used for training the models, while the scheduled data for the upcoming semesters are used for prediction. Under Figure 1 (A), the input $X^{k\times l}$ describes student-course features, where k is the number of input instances, and l is the number of features, such that $\{k, l\} \in \mathbb{N}$. The desired outputs consist of (1) student performance, represented as the total assessment grade G^k, i.e., $g_i = [0, 100]$, and (2) a set of factors $GV^{k\times o}$, such that $GV_{k,j} = \left\{gv_{k,j} : j \in \ 1, 2, \cdots, o\right\}, o \in \mathbb{N}$, and every target $gv_{k,j}$ must be typed over a quantitative value. We distinguish between student and course features by $\dot{X}$ and $\ddot{X}$, respectively.

Given an educational dataset, as described in Figure 1 (A), the main challenge in this paper is to predict values for G and GV that approximate the best mapping between X, G, and GV in the hypothesis space H, such that all the inner predictors belong to H, typically for input-instances that belong to the current and/or upcoming semesters. We address this problem in two sequential steps, illustrated in (B) and (C) of Figure 1. Firstly, we approximate precise values for G using a hybrid regression model (HRM). Then, we re-consider G as a feature besides the input-instances in X for training (MLSOM) an unsupervised neural network model (i.e., represented as self-organizing map) for approximating the fittest values for GV. To further clarify the illustrative steps (A), (B), and (C) in Figure 1, we abstractly describe the main top-level steps of our approach in algorithm 1. The underlying procedures and formulas implementing these steps are explained in detail in the following subsections.

A. THE HYBRID REGRESSION MODEL (HRM)

Multiple circumstances often influence student performance during their educational journey, some of which are strictly related to (e.g., types of courses and student's abilities, etc.). It is, therefore, challenging to rely on a single learning model to predict student grades G. Students usually have different characteristics and historical behaviors, and the use of a single model may lead to inferring imprecise predictions, which

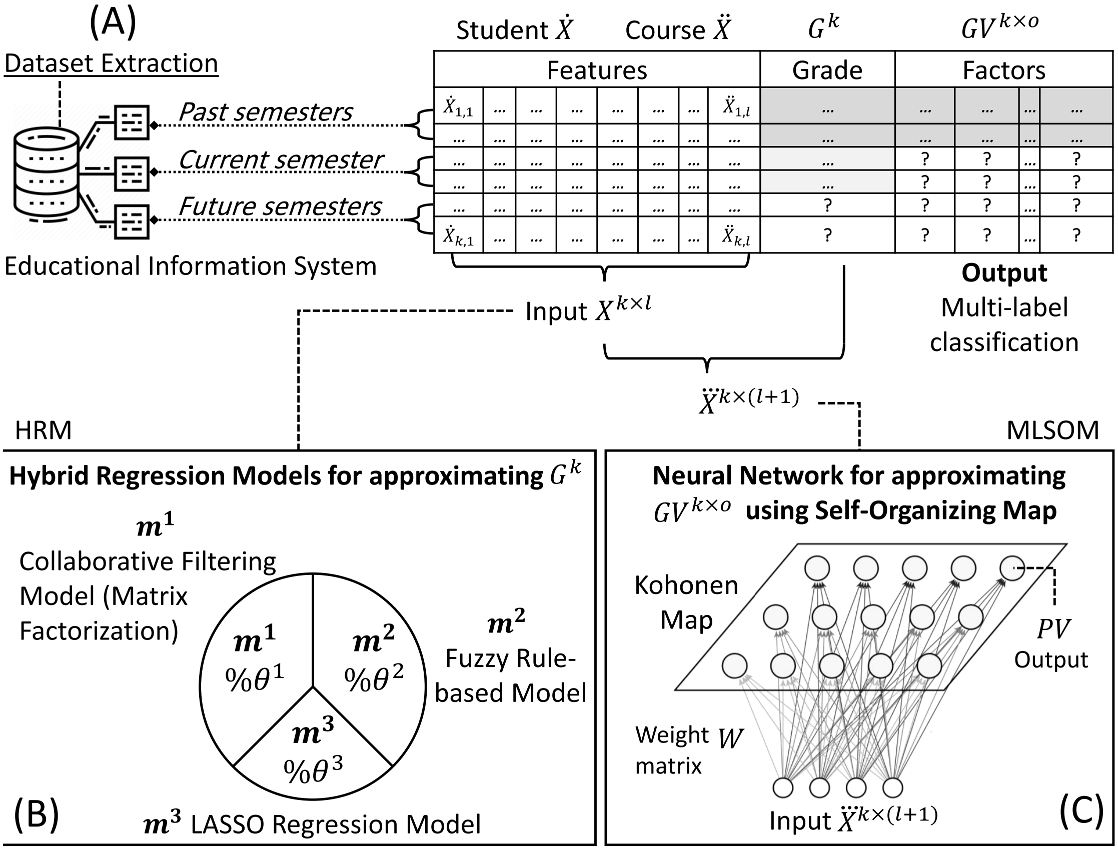


FIGURE 1. Overview of the proposed hybrid regression (HRM) and multi-label self-organizing map (MLSOM) models.

```
Algorithm 1: High-level steps for training and predicting student performance ($G$) and their associated factors ($GV$) using the proposed HRM and MLSOM models.
input  : $DS$: a clean educational dataset, as described
         in Figure 1 (A)
output: $G$ and $GV$: student grades and their associated
         factors
$DS^{train}, DS^{pred}$ ← Split ($DS$)
splitting $DS$ into $DS^{train}$ and $DS^{pred}$ depends on
time-series, such that the training data are extracted
from the past and the currently running semesters, while
the scheduled data for the upcoming semesters are used
for prediction.
HRM.train ($DS^{train}$)
G ← HRM.predict ($DS^{pred}$)
MLSOM.train ($DS^{train}$)
$DS^{pred}$.update ($G$)
GV ← MLSOM.predict ($DS^{pred}$).
```

could be a result of some obvious issues, e.g., cold-start and overfitting problems. Hence, to improve the accuracy of our predictor, we train not a single model but a hybrid model that allows one to combine different logical anticipations. The postulated hybrid regression is a combination of collaborative filtering model (denoted as m^1), fuzzy rule-based model (m^2) and Lasso linear regression model (m^3), as depicted in Figure 1 (B). Here each model has a specific weight θ to determine its influence according to student circumstances, such that the final decision by the hybrid prediction is expressed as:

$$HRM(x_i) = \sum_{n=1}^{3} m^n(x_i)\theta^n \tag{1}$$

where x_i is the input instance. We assume a weighted sum model to determine, firstly, the values for θ^1 and θ^2 for combining m^1 and m^2, respectively. Let $\dot{x}_i^{\varphi}$ and $\ddot{x}_i^{\varphi}$ denote the ratio of the completed courses by the student $\dot{x}_i$ and the ratio of teaching the course $\ddot{x}_i$ (i.e., the number of times the $\ddot{x}_i$ appears in X divided by k), respectively. We then calculate the first two weights for θ^1 and θ^2 as follows:

$$\theta^1 = \frac{\dot{x}_i^{\varphi}(1-\theta_3)}{(\dot{x}_i^{\varphi}+\ddot{x}_i^{\varphi})}, \theta^2 = (1-\theta^1-\theta^3) \tag{2}$$

For m^3, we consider its weight θ^3 as an important predefined parameter that has to be configured prior to the testing or the practical prediction stage.

1) COLLABORATIVE FILTERING MODEL

The principle behind our collaborative filtering model m^1 is to predict student performance by discovering the hidden patterns in the historical student-course relations in a neighbourhood-based. Collaborative filtering, in general, is a paramount approach for recommender systems [54], [55], particularly in e-commerce systems. It mainly aims to filter out user-item preferences based on the past ratings of comparable items and users. Therefore, the adoption of a collaborative filtering concept boosts m^1 to contribute better predictions for first-year students who are in their early academic stages (e.g., in the case when the model has insufficient information about specific students in X as its focus would be on discovering the behavior of the same courses, studied in the previous semesters) as the prediction here depends principally on the overall past relationships between, e.g., graduated students and their studied courses.

The most conventional collaborative filtering techniques are Nearest Neighbor [56] and Matrix Factorization [57], [58]. A series of methods have optimized both techniques, but we focus on the latter as it has demonstrated its efficiency in tackling some open problems, such as scalability, performance, and inaccurate predictions [59]. To be more precise, we train our collaborative filtering model m^1 based on implicit data using a promising non-negative matrix factorization technique (NFM) [60], [61] for regression predictions. Here, the implicit data means that each instance x_i, used for training, must have a valid value for g_i that links both student s and course c.

To clarify more, let GM denotes our student-course grading matrix, such that $GM \in \mathbb{R}^{|S| \times |C|}$, where $|S|$ and $|C|$ are the numbers of students and courses in X respectively. Then, the implicit data in GM are defined as:

$$GM_{sc} = \begin{cases} \mathrm{g} & \text{if } s \text{ has completed } c \text{ and } g \text{ is observed} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

In other words, each entry $(GM_{sc} : \; = \; 0)$ means that a student s has not completed a course c yet (i.e., potentially because c is allocated in the upcoming academic semesters for s), denoted as $\overline{GM}_{sc}$. This means that $\forall GM_{sc} \neq 0$, both s and c can be represented by partially observed vectors from the dataset X, such that $\dot{X} \widehat{=} (GM_{1c}, \cdots, GM_{kc}) \in \mathbb{R}^{|S|}$ and $\ddot{X} \widehat{=} (GM_{s1}, \cdots, GM_{sk}) \in \mathbb{R}^{|C|}$. Given a grading matrix GM, the matrix factorization technique aims to factorize GM into two non-negative matrices S and C, such that $GM \approx SC$. Here, S and C represent the latent factor vectors for a student s and a course c respectively, such that $S \widehat{=} \{\dot{X}_{1:k}\}$ and $C \widehat{=} \{\ddot{X}_{1:k}\}$. The $\{.\}$ is used as an expression for excluding the duplicated, e.g., student/course IDs. By learning S and C, one can estimate the unavailable values in $\overline{GM}_{sc}$ by using their inner product as follows:

$$\overline{GM}_{ij} = f(\dot{X}, \ddot{X} \mid S_i, C_j) = S_i^T C_j \tag{4}$$

The objective function for training S and C is based on optimising the distance d between GM and their matrix product SC, expressed by squared Frobenius norm[1] as follows:

$$d(GM, SC) = \frac{1}{2} ||GM - SC||^2 = \frac{1}{2} \sum_{i,j} (GM_{ij} - SC_{ij})^2 \tag{5}$$

2) FUZZY RULE-BASED MODEL

A fuzzy expert rule-based model, in its simple form, is an expert system but with more rules and fuzzy membership operations that go beyond the classical Boolean logic [62]. It fundamentally consists of two main components: a knowledge base, represented as $IF \cdots THEN$ rules, and an inference methodology for reasoning. Unlike the collaborative filtering model m^1, the logical rule-based model m^2 focuses on predicting student grades by analysing their past studied courses individually.

As discussed in section III, we have surveyed the most recent studies on student performance to identify the factors that influence student achievement and success of program learning outcomes. Through the factors gathered, we were able to build a sensible knowledge base, while considering a forward-chaining method (i.e., data-driven reasoning) in the inference engine. After determining the fuzzy sets of the distinct relevant factors (or attributes) that would have a direct impact on student's final grades in each course and then asking two education experts to estimate the output grade, we have constructed approximately 106 fuzzy rules. Regardless that the experts agree with their rules only, combining all of their rules made them imprecise. Consequently, we have used SkFuzzy[2] to determine better estimations of the fuzzy relationship between input and output variables of these rules. The generated base set of these computable rules is represented in Figure 2 as a cube matrix.[3] It takes three inputs, namely student Grade Point Average (GPA), selected set of courses, and GPA change within the last two semesters, to infer the type of grade for g_i as a defuzzified result.

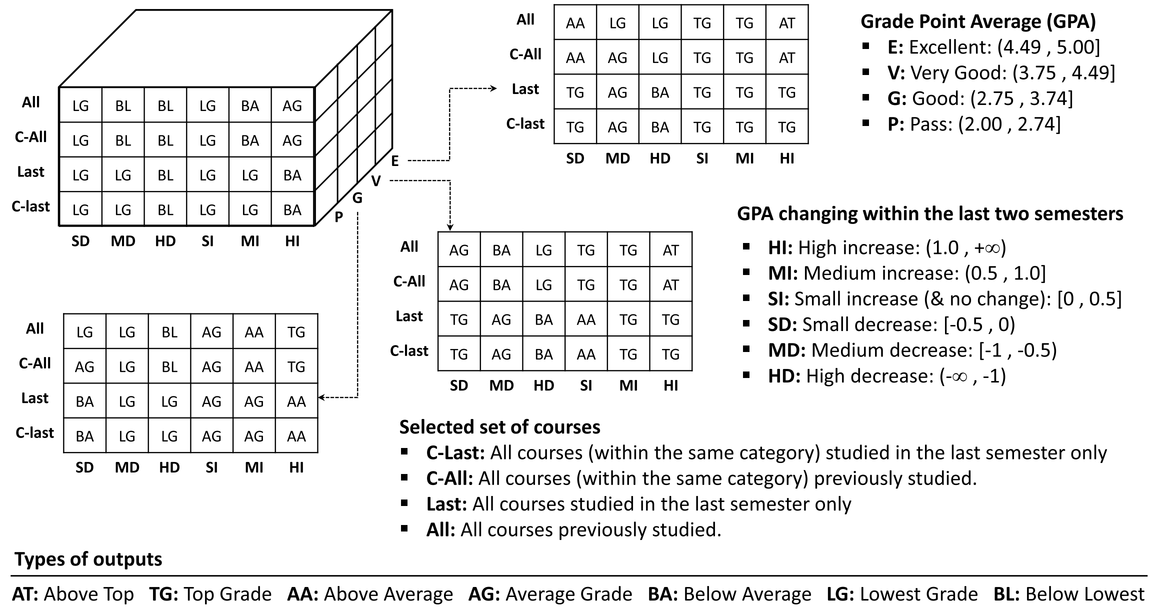
For instance, if a student s has a high GPA and that GPA has slightly improved, and the selected range of his/her courses is determined to include *all* the completed courses, the predictor m^2 in this case will return (TG), i.e., the highest overall grade obtained by s. Such rules can be defined as follows:

- **R1**: *if* course-difficulty-level is medium $\in$ $[0.4, 0.7]$
  and GPA $\in (4.49, 5]$
  then the selected range of courses is *All*
- **R2**: *if* the selected range of courses = All
  and GPA change = SI
  and GPA $\in (4.49, 5]$
  then the type of grade is *TG*

[1] The squared Frobenius norm is an extension of the Euclidean norm function https://scikit-learn.org/stable/modules/decomposition.html#nmf

[2] SciKit (skfuzzy 0.2) is a fuzzy logic toolbox written in Python https://pythonhosted.org/scikit-fuzzy/index.html

[3] This cube representation is known as a fuzzy associative memory (FAM) [63]



	SD	MD	HD	SI	MI	HI
All	LG	BL	BL	LG	BA	AG
C-All	LG	BL	BL	LG	BA	AG
Last	LG	LG	BL	LG	LG	BA
C-last	LG	LG	BL	LG	LG	BA

	SD	MD	HD	SI	MI	HI
All	AA	LG	LG	TG	TG	AT
C-All	AA	AG	LG	TG	TG	AT
Last	TG	AG	BA	TG	TG	TG
C-last	TG	AG	BA	TG	TG	TG

	SD	MD	HD	SI	MI	HI
All	AG	BA	LG	TG	TG	AT
C-All	AG	BA	LG	TG	TG	AT
Last	TG	AG	BA	AA	TG	TG
C-last	TG	AG	BA	AA	TG	TG

	SD	MD	HD	SI	MI	HI
All	LG	LG	BL	AG	AA	TG
C-All	AG	LG	BL	AG	AA	TG
Last	BA	LG	LG	AG	AG	AA
C-last	BA	LG	LG	AG	AG	AA

Grade Point Average (GPA)

- **E:** Excellent: (4.49 , 5.00]
- **V:** Very Good: (3.75 , 4.49]
- **G:** Good: (2.75 , 3.74]
- **P:** Pass: (2.00 , 2.74]

GPA changing within the last two semesters

- **HI:** High increase: (1.0 , +∞)
- **MI:** Medium increase: (0.5 , 1.0]
- **SI:** Small increase (& no change): [0 , 0.5]
- **SD:** Small decrease: [-0.5 , 0)
- **MD:** Medium decrease: [-1 , -0.5)
- **HD:** High decrease: (-∞ , -1)

Selected set of courses

- **C-Last:** All courses (within the same category) studied in the last semester only
- **C-All:** All courses (within the same category) previously studied.
- **Last:** All courses studied in the last semester only
- **All:** All courses previously studied.

Types of outputs

AT: Above Top **TG:** Top Grade **AA:** Above Average **AG:** Average Grade **BA:** Below Average **LG:** Lowest Grade **BL:** Below Lowest

FIGURE 2. Self-explanatory figure representing sliced cube matrix for our fuzzy rules.

3) LASSO REGRESSION MODEL

In m^3, we consider Lasso[4] regression model, an optimised type of linear regression [64]. Apart from the fact that our dataset X is entirely normalized, it describes continuous features only and does not contain any missing values. However, the probability of having useless features or data outliers subsists. Therefore, the use of Lasso model can enhance the overall accuracy of m^3 in two ways; (1) it simplifies the model by performing a feature selection latently, which neglects the useless features that have no significant influence; (2) it estimates sparse coefficients whilst adding a penalty as a regularizer, which prevents the model from being overfitted. More precisely, the goal of Lasso is to minimize [61]:

$$\underset{w}{\text{argmin}}\{\frac{1}{2k}||Xw - g||_2^2 + \lambda||w||_1\} \tag{6}$$

where w is the coefficient vector, $||.||_2$ is ℓ_2-norm, and λ is a constant that is normally set to 0.1. Once the best coefficient vector w is found, we can calculate the prediction for a given input X_i using a simple linear regression model described as:

$$m^3(x_i) = w_0 + \sum_{n=1}^{j} w_m(x_{i,n}) \tag{7}$$

where w_0 is the intercept.

B. MULTI-LABEL CLASSIFICATION FOR PREDICTING KEY FACTORS

Once the prediction of student performance (i.e., determined in G) is completed by the hybrid regression model, then the model examines the reasons why such performances took place. To a reasonable extent, this examination would reveal the key strengths and weaknesses behind the attainment of student learning outcomes by predicting values for each factor in $GV^{k\times o}$, see (C) of Figure 1. Unlike the usual single-label classification problem, where the goal is to create a predictor that classifies each input instance to a single class output (i.e., defined in a label containing a set of disjoint classes) at a time, our problem is probably more complicated as each input instance has to be classified into more than one class (i.e., factor) concurrently. More explicitly, our multi-label classification problem can be formulated as follows:

$$\underset{\ddot{X}\to gv}{\text{fit}} \; F(\ddot{X}) = (f_1(\ddot{X}), f_2(\ddot{X}), \cdots, f_o(\ddot{X})) \tag{8}$$

where $o > 1$ is the number of factors in GV. Here, at this stage, we treat G as an input feature in addition to X, expressed as $\ddot{X}$, i.e., $\ddot{X} \in X \cup G$. While the objective is to approximate the most fitted values for gv_i that map precisely $f : \ddot{x}_i \to gv_i$.

There has been an expanded body of research on using supervised learning techniques to address multi-label problems. Much of these works are Binary-Relevance based [65] on which the labelled classes are tackled independently. Towards a more dependable generalization, however, one should rely on Label-Powerset (defined in [50]) and take the correlation between labeled classes into consideration. In our research, it is of great importance to recognize the relationship between gv_i, which principally can be reached by clustering related sets of labels. To this end, we apply a Self-Organizing Map (SOM) [66], an effective Label-Powerset based method, through the use of neural networks in an unsupervised learning manner. SOM, in general, is a neighborhood-preserving approach based on competitive

[4] LASSO stands for Least Absolute Shrinkage and Selection Operator

learning [66]. It attempts to maintain the topological associations of a high-dimensional input space by mapping it to a low-dimensional space (the so-called Kohonen layer). We refer to the second part of our model as Multi-Label Self Organizing Map model (MLSOM).

Given a two-dimensional Kohonen network, consisting of some neurons, where each neuron n is associated with a weight vector w^n and a prototype vector pv^n (i.e., used for the output prediction). The dimensions of these two vectors (i.e., w^n and pv^n) must be identical to the input x_i and the output gv_i vectors, respectively. The underlying training process of SOM can be concisely described as follows:

1) initialise the weight w^n for all neurons randomly;
2) find a neuron n^{win} that precisely fits the given input pattern x_i, so-called winner neuron;
3) determine all the neighborhood neurons around the n^{win}, denoted as $n^{excited}$;
4) update the weight of all w^n associated with $n^{excited}$; and
5) iterate the steps 2-4 multiple (epoch) times with different inputs.

The prediction process (by giving.e.g. testing instances) begins by also (1) determining the best mapping to the n^{win} as well as the surrounded $n^{excited}$, and then (2) averaging the labels associated with the training input instances (i.e., mapped to n^{win} and $n^{excited}$ after the training) onto pv^n.

The closely related works, in terms of using SOM for multi-label prediction, are [50], [51]. We are mostly in line with [51] in (1) determining the n^{win} for a test input instance $\dddot{x}_i$ by minimising the Euclidean distance, see (9) as well as measuring the lateral distance between n^{win} and all its neighboring neurons $n^{excited}$ using the Gaussian function, see (10). Nevertheless, we differ with them in that our prototype vector pv^n is non-binary, and more importantly, we assume a weighted average method when calculating the values for pv^n. In other words, we are not treating all the associated training instances with both n^{win} and $n^{excited}$ evenly, but rather some training instances are more important than the others.

Let us be more explicit about training our neural network. Given an input $\dddot{x}_i$ at a certain epoch iteration, we compute its distance to all neuron weights w^n, such that the minimum distance determines the n^{win}, expressed by the Euclidean function as follows:

$$d_n(\dddot{x}_{ij}, w^n) = \underset{d}{\operatorname{argmin}} ||\dddot{x}_{ij} - w^n|| = \sqrt{\sum_{t=1}^{j+1} (\dddot{x}_{it} - w_t^n)^2} \tag{9}$$

where $d(.,.)$ returns the index of n^{win}. Then, its lateral distance with the neighboring neurons $n^{excited}$ are determined by the Gaussian function as follows:

$$h_{n^{win},n^{excited}} = exp\left[-\frac{d^2_{n^{win},n^{excited}}}{2\sigma^2}\right] \tag{10}$$

where the σ parameter defines the range of neighbourhood excited neurons. Next, each participated excited neuron in the learning process will get updated as:

$$\Delta w^{new}_{n^{excited}} = w^{old}_{n^{excited}} + \eta h_{n^{win},n^{excited}} \left[\dddot{x}_i - w^n\right] \tag{11}$$

where the η parameter defines the learning rate. The basic algorithms and fundamental steps of training an SOM with making a prediction are discussed in [50].

Saini *et al.* [51] has demonstrated the influence of including the mapped training instances with both n^{win} and $n^{excited}$ when averaging their classes' values onto pv^n. We apply their effective procedure and extend it by introducing a weighted mean formula for a more accurate prediction. Let $gv^{training}$ be the mapped training instances with n^{win} and $n^{excited}$ that are determined by an input test instance x_i^{test}, such that $gv^{training} \in gv$. Here, as each output $gv_i^{training}$ is determined by a neuron (i.e., either n^{win} or $n^{excited}$), we assume its level of importance by the lateral distance (i.e., defined in (10)), such that the smaller the distance determined, the higher the weight is assigned. The overall weighted average after completing the training process besides computing the prediction values for pv^n are explicitly introduced in Algorithm 2.

Algorithm 2: Weighted average for computing pv_t^n

```
input  : $X^{train}$ and $GV^{train}$: input instances and
         multi-label part from the training dataset
         $x^t$: an input test instance
output: $pv_t^n$: the prediction vector for the input $x^t$

$n_t^{win} \leftarrow$ findWinner($x^t$)
$n_t^{excited} \leftarrow$ getNeighbor($n_t^{win}$)
$factors[] \leftarrow \emptyset$
$nDistances[] \leftarrow \emptyset$
for $i \leftarrow 1$ to $|X^{train}|$ do
    if ($x_i^{train} \leftarrow$ isMapped($n_t^{win}||n_t^{excited}$)) then
        factors.append($gv_i^{train}$)
        nDistances.append($h_{n_t^{win},n_t^{excited}}$)
        see (10) for computing the distance between $n_t^{win}$
        and $n_t^{excited}$
$\widehat{nDistances}$ = normalise($nDistances$)
$weightedFactor = \widehat{nDistances} \times$ factors
$pv_t^n$ = getColumnWiseSum($weightedFactor$)
```

V. EXPERIMENTAL RESULTS AND ANALYSIS

To investigate the applicability of our approach, we conduct several experiments intending to address the following questions:

- What is the computational efficiency of the proposed hybrid regression method in its ability to predict student performance at different academic levels of study?
- How does the hybrid regression parameter θ^n affects the accuracy of predicting student performance?
- Does our proposed optimization for multi-label prediction helps to outperform the state of-the-art implicit self-organizing map methods?

As proof of concept, we implemented the hybrid regression and multi-label prediction (MLSOM) models presented in IV using Python. Codes and materials for replicating our experiments are available from the accompanying website for this paper.[5]

A. EXPERIMENTAL SETUP

1) SOURCE OF DATASETS

We experimented with seven publicly available datasets. The numerical descriptions of these datasets are presented in Table 3. The first two (UD[6] and OULAD[7]) are academic datasets, which are used for assessing the whole approach, i.e., includes both the hybrid regression and MLSOM models.

TABLE 3. **Numerical descriptions of the evaluation datasets.**

	# Inst.	**# Feat.**	**# Labe.**	**Card.**	**Dens.**	**# Dist.**
UD OULAD	206700	16	7	5.167	0.326	128
cal500	502	68	174	26.04	0.150	502
birds	645	258	19	1.014	0.053	133
emotions	593	72	6	1.869	0.311	27
flags	194	10	7	3.392	0.485	54
scene	2407	294	6	1.074	0.179	15
yeast	2417	103	14	4.237	0.303	198

Instances - **Feat**ures - **Labe**ls (Factors) - **Card**inality - **Dens**ity - **Dist**inct

TABLE 4. **The number of students and courses used in training and validating our models based on UD and OULAD datasets.**

Current semester			**Upcoming semester**	
Level	# Students	# Completed S-C	Next Level	# Registered S-C
Entry	390	0	1	2680
1	390	2682	2	2670
2	390	5354	3	2676
3	390	8041	4	2673
4	390	10714	5	2681
5	390	13352	6	2667
6	390	16056	7	2684
7	390	18653	8	2673
8	700 *	38420	–	–
	3820	**113272**		**21404**

S-C : Students-Courses
* : Graduated students

Specifically, the UD provides the main features of 3820 students and 56 courses with their actual final grades, studied in an academic program that has eight levels of study, where each level consists of 7 courses (i.e., 7 × 8 = 56). The average number of courses taken by each student is 6 per semester, see the numerical details of students and courses, broken down into the eight academic levels, in Table 4. Here, student levels are ordered chronologically by annually academic semesters, and all data defined as *current semester*, including courses completed by graduated students, are used for training our models. While the scheduled data for the next semester that is specified as *upcoming semester* are used for validation. Besides classifying all the 56 courses into eight levels of study, they are also categorized into three types (i.e., mandatory at the faculty/department levels and elective). We preprocessed and combined UD with OULAD to generate random multi-label factors as each $gv_{i,j} = [0, 5]$, based on different types of assessments for each student and courses.

We implemented a data-processing tool to clean, normalise and simulates the calculation of student's GPA after completing each semester. Consider (A) of Figure 1, the split of the generated dataset into training and testing parts is carried out according to the chronological order of student levels. This means all input instances belonging to students at the final level (8) will always be part of the training set. With UD and OULAD datasets, our focus is on predicting student's grades and factors for only the next upcoming semester.

The rest of the datasets (Cal500, Birds, Emotions, Flags, Scene, and Yeast) utilized in our experiments are associated with non-academic domains, including music, audio, images, biology, but used as a benchmark for evaluating the performance of our MLSOM. As mentioned, the required academic datasets for this approach are often hard to find in publicly available sources. Besides, all the related studies that we are aware of, conducted at the university level, did not offer their datasets for experimental replication, probably due to data privacy restrictions.

The statistical characteristics of the used non-academic datasets can be found in [50], [67]. The types of these datasets are valid according to our declared dataset, described in Figure 1 (A). In particular, these benchmarked datasets contain numerical multi-label outputs that are abstractly representative of different rates of influential factors. Consequently, they can be utilized for performance evaluation in general terms. Moreover, these datasets seem appropriate as they offer various important characteristics, such as data density, cardinality, and distinct [50].

2) BASELINES AND PARAMETER SETTINGS

Although our hybrid regression model depends on both LASSO regression [64] and non-negative matrix factorization technique (NFM) [60], [61], they can also be utilized as a baseline for evaluating the overall accuracy. For MLSOM, we consider the main relevant competitor approaches: SOM-MLL [50] and ML-SOM [51] as a baseline for evaluation.

For making the experimental results comparable on all datasets, configured as approximately 81% training and 19% validation settings, we set the random number generator in Python using, e.g., $NumPy.random.seed(0)$. This is to ensure generating the most similar random weights for each run. Furthermore, the reported figures and results are averaged over ten runs as each run begins with a random seed value.

[5] https://github.com/IU-Distinction-Project/

[6] University Dataset (UD): https://www.kaggle.com/ananta/student-performance-dataset

[7] Open University Learning Analytics Dataset (OULAD): https://www.kaggle.com/rocki37/open-university-learning-analytics-dataset

TABLE 5. General settings of the main parameters. For θ^3, we have estimated its sensitivity with the performance of our hybrid model and found its best setting value is ranged between (0.35, 0.45). Since no formal rule for setting the k-fold parameter, we have trained the LASSO model on 5-fold, which is a reasonable choice with small to medium dataset (i.e., it allows the size of each splitting group of data to be statistically representative for minimizing the potential bias of the LASSO model). Further, no formal rule for configuring the SOM topology, but within the context of our approach, 36 neurons is a reasonable size for clustering all our student-course instances. To start with a good point on training our MLSOM, we set $\eta = 0.1$ (i.e., the traditional default value) and configured σ to begin with covering 75% of all neighbor neurons.

Parameter	
MLSOM parameter (θ^3)	0.35
Cross-validation for LASSO Model (m^3)	5-fold
SOM Topology	Square
Number of neurons	$6 \times 6 = 36$
Learning rate (η)	0.1
Neighbourhood radius (σ) begins with the number that covers 75% of all neighbour neurons	4.5

The applied configurations and settings of the main parameters are presented in Table 5.

3) EVALUATION METRIC

We used the root mean squared error (RMSE) to precisely assess the closeness of model predictions (i.e., student performances and factor ratings) as compared to the actual output values. In this paper, the well-known RMSE is a reasonable choice since the hybrid regression model is itself a competitor to m^1, m^3, and m^2, besides that distinguishing the slight variations between the predicted values by these models are statistically significant. The RMSE is defined as

$$RMSE = \sqrt{(\frac{1}{|V|})\sum_{i=1}^{|V|}(g_i - m_i^n)^2} \tag{12}$$

where V denotes the validation set. The same metric is also used for measuring the accuracy of the predicted values in gv_i.

B. EFFICIENCY OF THE PROPOSED HYBRID REGRESSION METHOD (HRM)

The performance (RMSE) comparison of m^1, m^2, m^3, and HRM are illustrated in Figure 3 and Figure 4. The overall prediction accuracy of these four predictors looks resembling based on UD and OULAD datasets. However, our proposed HRM method clearly outperforms the competitors over almost all the academic levels, as depicted in Figure 3. Here, we can observe that m^2 has begun with a poor performance (Note: its predictions starts from the second academic level), but it has maintained to improve as the levels get progressed until around level seven. This is relatively reasonable because m^2 depends principally on examining each student's performance at their previous levels individually. Meanwhile, m^1 has fluctuated over all levels and became the worst in the last four levels. Interestingly, the performance of HRM demonstrates its stability in predicting values for G effectively despite the fluctuated changes in the performance of m^1 and m^2. This stability is due to the proposed idea, in (2), for adjusting the values of the weights (i.e., θ^1 and θ^2) dynamically according to the student and course features.

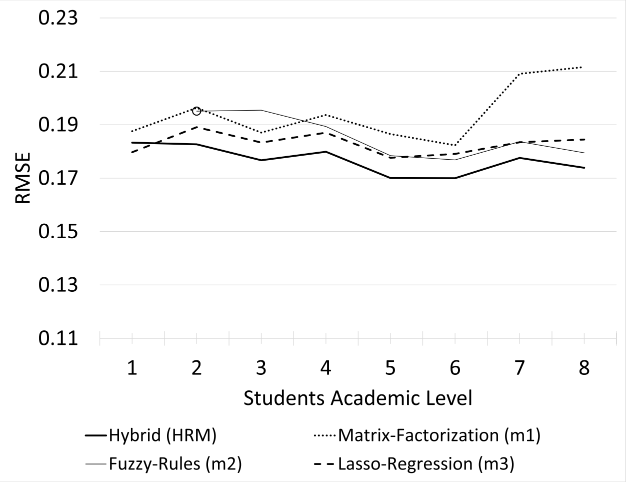


FIGURE 3. RMSE comparison of m^1, m^2, m^3, and HRM w.r.t the eight academic levels on UD and OULAD datasets (A lower RMSE value indicates better performance.)

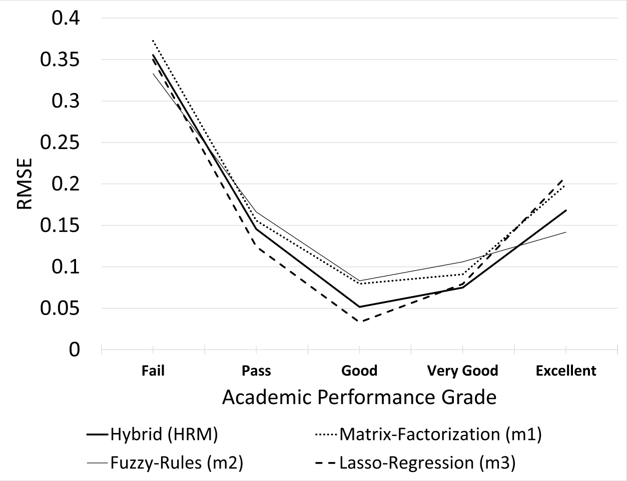


FIGURE 4. RMSE comparison of m^1, m^2, m^3, and HRM w.r.t student's performance on UD and OULAD datasets.

Figure 4 illustrates the RMSE differences between the competitors, including ours, with respect to the students performance in each course (i.e., the testing data are grouped by the actual G values into five grading scales). It demonstrates the effectiveness of each predictor in accordance with the standard grading scales (i.e., Fail, Pass, Good, Very good, and Excellent). While also the performance disparity between the competitors appears relatively insignificant, it concludes that none of these predictors can output the best values for G in all cases. Despite this, the results indicate the suitability of our proposed HRM as it would, at least, give the second to the best performance (if not the first) in most cases.

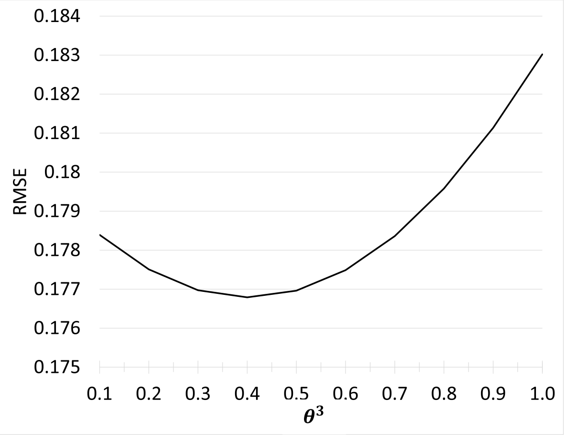


FIGURE 5. Impact of parameter θ^3 on the performance of hybrid regression model (HRM).

To give insight into how our HRM behaves in combining m^1, m^2, and m^3, we have conducted a sensitivity analysis on the parameter η^3. We measured the performance of HRM after setting η^3 with different weights between [0, 1]. Here, If $\eta^3 = 0$, the HRM will combine the predicted G from only m^1 and m^2. In contrast, If $\eta^3 = 1$, it means that HRM will neglect m^1 and m^2, and therefore its prediction will be identical to the prediction given by Lasso regression model (m^3). The result is reported in Figure 5, which shows global convexity behaviors around the interval of (0.35, 0.34), depicting the range of the best values for the parameter η^3.

C. EFFECT OF OPTIMIZING OUR MULTI-LABEL PREDICTION USING A WEIGHTED AVERAGE

Here, we have performed two experiments to explore the productive model capacity of our MLSOM besides evaluating its performance (RMSE) in comparison with SOM-MLL [50] and ML-SOM [51]. As mentioned earlier, the essential difference between these two baselines is that SOM-MLL focuses on producing a prototype vector pv^n for each input instance $\ddot{x_i}$ by averaging label vectors of training instances that are mapped with the winning neuron (i.e., n^{win}) only. Whereas, ML-SOM expands the scope of label vectors by taking also the mapped training instances with neighboring neurons (i.e., $n^{excited}$) into consideration. In MLSOM, we consider the latter method but without treating all the mapped training instances evenly as we assume a weighted mean method for computing the prototype vector pv^n.

Since one of the prime parameters in determining the efficiency of ML-SOM and MLSOM is associated with the neighbourhood radius around n^{win}, we have conducted a sensitivity analysis on the parameter σ using UD and OULAD datasets. The result is reported in Figure 6, which shows the advantage of MLSOM over both SOM-MLL and ML-SOM. Here, we shed light on that the best mapping units (i.e., n^{win}) by the three competitors are identical for all input tests, and that σ has no impact on SOM-MLL as it neglects the neighboring neurons. Meanwhile, the performance of ML-SOM and MLSOM are very sensitive to σ.

With expanding the neighborhood (i.e., leads to computing pv^n from more training instances), ML-SOM appears to give an excellent performance up to approximately $\sigma = 3/10$, and becomes significantly worse after expanding this limited radius. In contrast, the highly effective performance of MLSOM starts to drop at a larger radius (i.e., at approximately $\sigma = 6/10$), and more importantly, it did not produce worse performance than SOM-MLL in all radius. This observation can conclude that the inclusion of neighboring neuron information would either (1) improve the performance of MLSOM or (2) at least have insignificant negative import on the performance as compared to SOM-MLL. The reason behind this is due to the idea of using the weighted mean as the larger the neighboring neuron deviates from the winning neuron, the less important it becomes when computing pv^n.

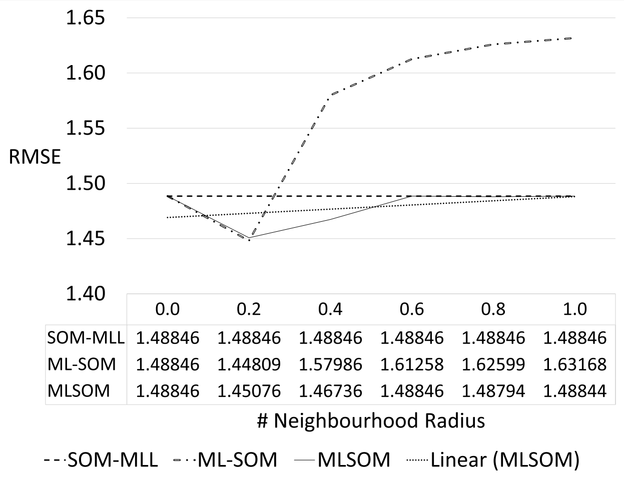


FIGURE 6. RMSE comparison of SOM-MLL [50], ML-SOM [51], and our MLSOM w.r.t different neighbourhood radius (σ) on UD and OULAD datasets.

TABLE 6. RMSE comparison of SOM-MLL [50], ML-SOM [51], and our MLSOM on the non-academic datasets. The validation was performed based on $\sigma = 1.5$.

	SOM-MLL [50]	ML-SOM [51]	Our MLSOM
cal500	0.34	0.32 *	0.32 *
birds	1.29	1.23 **	1.25
emotions	0.45	0.46	0.43 **
flags	1.47	1.61	1.46 **
scene	0.31 **	0.34	0.32
yeast	0.56	0.42 **	0.43

* indicates the best performance

In the second experiment, besides the performance comparison, we were interested in exploring under which datasets characteristic the competitors can produce satisfactory performance. Table 6 presents the RMSE of SOM-MLL, ML-SOM and MLSOM on the six datasets, described in Table 3 and based on the configuration shown in Table 5. Regardless of the apparently insignificant differences in the overall performance of the three classifiers, the result demonstrates that none of these classifiers, including ours, does output the best values for the factors GV in all cases.

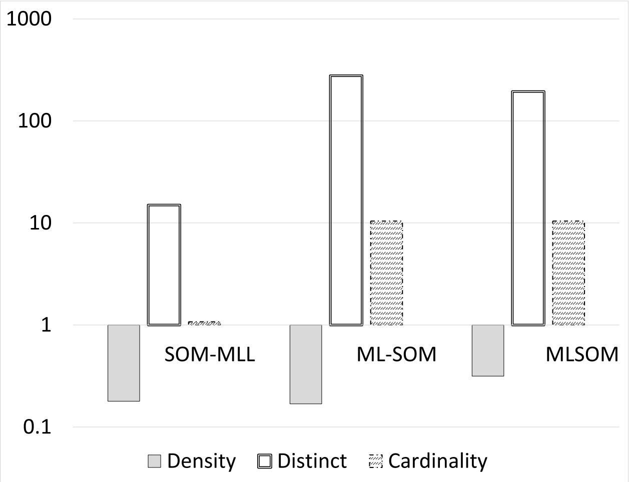


FIGURE 7. Comparison of SOM-MLL [50], ML-SOM [51], and our MLSOM w.r.t different characteristics of the datasets.

We try out to understand the reason for such optioned performance discrepancies through reviewing the characteristics of the datasets. Here, we found that the label density/cardinality, as well as the number of distinct labels, play a significant role in influencing the overall performance of these classifiers. We arrive at a conclusion, described in Figure 7, as SOM-MLL is apparently suitable with a low level of label cardinality. While, with a high level of label distinct, both ML-SOM and MLSOM would produce better performance than SOM-MLL. It also appears that our MLSOM would be a better choice with a low level of density.

TABLE 7. Statistical T-test results based on the RMSE measured in Figure 3 and Figure 6.

HRM	t-value	p-value
Matrix Factorization (m^1)	-4.11946	.000521
Lasso Regression (m^3)	-2.69393	.008732
MLSOM		
SOM-MLL [50] (*significant at p <.10*)	-1.51415	.080467
ML-SOM [51]	-2.652	.012115

VI. DISCUSSION

A. STATISTICAL HYPOTHESIS TEST CONSIDERATION

In order to test whether the measured performances (RMSE) of the proposed HRM and MLSOM are significant or subject to random error, we have conducted statistical hypothesis t-tests, based on two independent means. The test (t-value) and the probability (p-value) are reported in Table 7, which show that our results are statistically significant. These values were calculated using an online tool,[8] a significance level of 5%, and a one-tailed hypothesis (i.e., there is a rational expectation that the proposed HRM and MLSOM would give a poor performance, compared to their competitors).

Assuming the null hypothesis H_0 that *"there is no difference between the performance (RMSE) achieved by HRM/MLSOM and their baseline competitors"* is true, which will be rejected if $p < .05$. Here, due to random error, H_0 indicates that there are .000521, .008732, .080467, and .012115 probabilities of achieving the same performance by HRM and MLSOM as compared with m^1, m^3, SOM-MLL, and ML-SOM, in respectively. The null hypothesis is thus rejected, and indeed our proposed HRM and MLSOM models can provide extremely competitive performance as compared with the other approaches.

B. PRACTICAL IMPLICATIONS

Our research shows that it is possible to go beyond the simple task of predicting student performance that does not reveal the reasons behind particular student achievements. The majority of existing research studies [13] attempted to create predictive models of student performance only. While this might help program leaders understand the overall success of their educational programs and identify students who are at risk of failing or dropping out, it certainly does not highlight the possible causes and issues that inhibit efficient student learning and attainment of student learning outcomes.

Our detailed methodology combines various predictive techniques, e.g. regression based models and unsupervised learning models, to improve the accuracy of student performance predictions. More importantly, the proposed hybrid model takes an extra step of explaining the given outcomes, i.e., explanatory modelling in terms of factor predictions, based on the available inputs. The implementation and testing datasets are made available for other researchers to explore and investigate. Our flexible model has the capacity to account for several types of factors, as input, at the same time, ranging from student demographics, psychomotor characteristics, socio-economic status, course grades, and grade point averages, $\cdots$, etc. Other factors may be easily included in our proposed model.

Today's higher education institutions store a wealth of student data with regard to student historical (e.g., pre-university) and current performance as well as their circumstances, yet this is not exploited correctly. Our model is one motif for the higher education sector to move towards learning analytics to enhance the quality of their offerings and prepare graduates who meet the job market demands.

C. THREATS TO VALIDITY AND RESEARCH LIMITATIONS

Concerning the feasibility of utilizing the proposed hybrid/multi-label classifier models and the soundness of the conducted experiments, we discuss the main threats that might potentially affect the internal and external validity of our approach. In internal validity (i.e., pertains to the issues that may affect the achieved results), the threats might include the biases in setting up the experiments. Whereas, a potential threat to external validity (i.e., focusing on investigating the range of generality) is represented by carrying out experiments on non-real datasets.

[8] https://www.socscistatistics.com/tests/studentttest/default2.aspx

For unintended biases issue that might be introduced when we configured our experiments, we have reduced this threat by reporting on the results that are averaged over ten similar configured runs. Since each run may result in different measurements due to the randomization when initializing weight coefficients, we also address this issue by setting the Random Number Generator in Python (using $NumPy.random.seed(0)$) to ensure generating the most similar random weights for each run (particularly when we measured the sensitivity of θ^1 and σ).

Regarding the generality of our approach, a significant challenge to undertake predictive modelling in the domain of teaching and learning is finding appropriate and comprehensive datasets that cover student achievements throughout many years and for different degrees. We used the best open datasets that we could find from public datasets (i.e., Kaggle), which introduces a sampling bias. Obviously, such limited datasets do not include all possible factors that may impact student achievements. For example, student engagement is unavailable and not accounted for despite its undeniable importance. The datasets represent the performance of specific majors, which may differ from the performance of students pursuing other degrees, e.g., Psychology. Moreover, we have validated our model on non-academic datasets, which may portray patterns that differ from real educational datasets.

Furthermore, the predicted student success is measured using GPA, which is calculated using final course grades. Nonetheless, academic success has been shown to contain several measurements [52]. In other words, student academic success is a multi-concept construct that can be viewed from different angles. It is unclear whether the model will be able to predict, with a similar accuracy, other indicators of academic success, for example standardised test scores and percentage of student outcome attainment, especially when multiple indicators are used together to infer student success. Indeed, this is a research task to explore in the future. Moreover, we have not carried out additional testing with different datasets to confirm the reliability of models.

Therefore, we could not claim our approach's generality, but it may be necessary to tune the hybrid/multi-label classier models to fit other educational settings and datasets. Future work will explore the mentioned limitations in more inclusive detail using real academic datasets.

VII. CONCLUSION AND FUTURE WORKS

We proposed and validated an intelligent hybrid approach for predicting student academic grades while determining the dominant factors that led to the predicted performance. This approach will help program leaders in identifying the main strengths and weaknesses behind the academic achievements of their students. Such understanding is essential for implementing the necessary interventions to improve student performance. We have demonstrated that integrating different predictive modeling techniques leads to high accuracy predictions compared to using a single approach. We have also detailed our methodology to achieve this integration so that other researchers can replicate and test our model with their educational datasets. The proposed approach is flexible and extendable to include any types of factors judged to be relevant to the program at hand. Below we suggest two research directions that we believe are worth pursuing in the future.

- Hybrid Approach Optimization: our empirical investigation on seven different datasets demonstrated that the proposed hybrid approach has the capacity to produce competitive performance when compared with related methods. In particular, we showed the stability of the hybrid regression model (HRM) in predicting student performance and the low sensitivity of MLSOM to the neighborhood radius when predicting influential multi-label factors. However, conducting a series of follow-up experiments based on comprehensive real datasets and user studies are still required. We encourage future research to define the best thresholds for distinguishing the high/low label density and cardinality in the datasets under study to apply the best multi-label classifier according to its characteristics.
- Prediction of the Attainment of Program Learning Outcomes: our review of past works showed an overarching necessity to predict student attainment of learning outcomes since they may represent a better indicator of program success, as well as the acquired student knowledge and skills than mere course grades or GPAs. Assessment of educational programs measures student performance and collects evidence about the student learning experiences, which are later used, as feedback, to improve the quality of programs purposefully. However, program assessment is no simple endeavor [68]. The majority of program assessment systems are neither computerized nor can predict student attainment of student learning outcomes. Program assessment tools rarely embed machine learning or mining techniques in the functionalities and reports they produce [69], which restrains program leaders from discovering the factors that lead to high or low levels of student performance [68]. In our judgment, anticipating program weaknesses would help to recommend improvement actions promptly.

  Program assessment goes beyond student scores to give an overall judgment of how well the graduates attain the knowledge, skills, and competencies needed by the job market. However, it does not pinpoint the enabling factors or inhibitors that could impact student learning [68]. The literature stipulates that a range of factors influence student academic performance and satisfaction. These factors range from instrumental characteristics, such as time management [22], high school grade, and class attendance [24], to psychological characteristics [23], such as test anxiety, motivation, and self-esteem. However, the impact of such factors on student outcomes varies from one program to another, even within the same university. Developing an intelligent learning model that predicts the attainment of program outcomes, given the variability of programs and student circumstances, is a promising and useful future research agenda for higher education.

REFERENCES

[1] A. M. Morcke, T. Dornan, and B. Eika, "Outcome (competency) based education: An exploration of its origins, theoretical basis, and empirical evidence," *Adv. Health Sci. Edu.*, vol. 18, no. 4, pp. 851–863, Oct. 2013.

[2] A. Namoun, A. Taleb, M. Al-Shargabi, and M. Benaida, "A learning outcome inspired survey instrument for assessing the quality of continuous improvement cycle," *Int. J. Inf. Commun. Technol. Edu.*, vol. 15, no. 2, pp. 108–129, Apr. 2019.

[3] L. M. B. Manhães, S. M. S. da Cruz, and G. Zimbrão, "Towards automatic prediction of student performance in STEM undergraduate degree programs," in *Proc. 30th Annu. ACM Symp. Appl. Comput. (SAC)*, 2015, pp. 247–253.

[4] J. McFarland, B. Hussar, J. Zhang, X. Wang, K. Wang, S. Hein, M. Diliberti, E. F. Cataldi, F. B. Mann, and A. Barmer, "The condition of education 2019 (NCES 2019–144)," U.S. Dept. Educ., Nat. Center Educ. Statist., Washington, DC, USA, Tech. Rep. NCES 2019-144, Oct. 2020. [Online]. Available: https://nces.ed.gov/pubsearch/pubsinfo.asp?pubid=2019144

[5] J. Bustamante. (Nov. 2019). *College Dropout Rate [2020]: by Year + Demographics*. EducationData.Org. [Online]. Available: https://educationdata.org/college-dropout-rates/

[6] A. Ahadi, R. Lister, H. Haapala, and A. Vihavainen, "Exploring machine learning methods to automatically identify students in need of assistance," in *Proc. 11th Annu. Int. Conf. Int. Comput. Edu. Res.*, 2015, pp. 121–130.

[7] A. Mueen, B. Zafar, and U. Manzoor, "Modeling and predicting students' academic performance using data mining techniques," *Int. J. Mod. Edu. Comput. Sci.*, vol. 8, no. 11, p. 36, 2016.

[8] 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.

[9] C. Brooks and C. Thompson, "Predictive modelling in teaching and learning," *Handbook Learn. Anal.*, pp. 61–68, 2017.

[10] A. M. Shahiri, W. Husain, and N. A. Rashid, "A review on predicting student's performance using data mining techniques," *Procedia Comput. Sci.*, vol. 72, pp. 414–422, Jan. 2015.

[11] M. Pandey and S. Taruna, "A comparative study of ensemble methods for students' performance modeling," *Int. J. Comput. Appl.*, vol. 103, no. 8, pp. 26–32, Oct. 2014.

[12] A. Satyanarayana and M. Nuckowski, "Data mining using ensemble classifiers for improved prediction of student academic performance," in *Proc. Spring Mid-Atlantic ASEE Conf.*, 2016.

[13] A. Hellas, P. Ihantola, A. Petersen, V. V. Ajanovski, M. Gutica, T. Hynninen, A. Knutas, J. Leinonen, C. Messom, and S. N. Liao, "Predicting academic performance: A systematic literature review," in *Proc. Companion 23rd Annu. ACM Conf. Innov. Technol. Comput. Sci. Edu.*, 2018, pp. 175–199.

[14] B. N. Gatsheni and O. N. Katambwa, "The design of predictive model for the academic performance of students at University based on machine learning," *J. Electr. Eng.*, vol. 6, no. 4, pp. 229–237, Jul. 2018.

[15] H. Aldowah, H. Al-Samarraie, and W. M. Fauzy, "Educational data mining and learning analytics for 21st century higher education: A review and synthesis," *Telematics Informat.*, vol. 37, pp. 13–49, Apr. 2019.

[16] K. L.-M. Ang, F. L. Ge, and K. P. Seng, "Big educational data & analytics: Survey, architecture and challenges," *IEEE Access*, vol. 8, pp. 116392–116414, 2020.

[17] M. G. Kavitha and D. L. Raj, "Educational data mining and learning analytics–educational assistance for teaching and learning," 2017, *arXiv:1706.03327*. [Online]. Available: http://arxiv.org/abs/1706.03327

[18] W. Deze, "Application of large data mining technology in colleges and universities," in *Proc. 2nd Int. Conf. Big Data Res.*, 2018, pp. 86–89.

[19] A. I. Adekitan and O. Salau, "Toward an improved learning process: The relevance of ethnicity to data mining prediction of students' performance," *Social Netw. Appl. Sci.*, vol. 2, no. 1, p. 8, 2020.

[20] A. I. Adekitan and E. Noma-Osaghae, "Data mining approach to predicting the performance of first year student in a university using the admission requirements," *Edu. Inf. Technol.*, vol. 24, no. 2, pp. 1527–1543, 2019.

[21] C. Lei and K. F. Li, "Academic performance predictors," in *Proc. IEEE 29th Int. Conf. Adv. Inf. Netw. Appl. Workshops*, Mar. 2015, pp. 577–581.

[22] N. Talib and S. S. Sansgiry, "Determinants of academic performance of University students," *Pakistan J. Psychol. Res.*, vol. 27, no. 2, pp. 265–278, 2012.

[23] B. Gębka, "Psychological determinants of university students' academic performance: An empirical study," *J. Further Higher Edu.*, vol. 38, no. 6, pp. 813–837, 2014.

[24] S. Sothan, "The determinants of academic performance: Evidence from a cambodian university," *Stud. Higher Edu.*, vol. 44, no. 11, pp. 2096–2111, 2019.

[25] Z. Shu, Q.-F. Qu, and L.-Q. Feng, "Educational data mining and analyzing of student learning outcomes from the perspective of learning experience," in *Proc. Educ. Data Mining*, 2014.

[26] A. Hernández-Blanco, B. Herrera-Flores, D. Tomás, and B. Navarro-Colorado, "A systematic review of deep learning approaches to educational data mining," *Complexity*, vol. 2019, pp. 1–22, May 2019.

[27] A. Polyzou and G. Karypis, "Grade prediction with models specific to students and courses," *Int. J. Data Sci. Analytics*, vol. 2, nos. 3–4, pp. 159–171, Dec. 2016.

[28] S. J. H. Yang, O. H. T. Lu, A. Y. Q. Huang, J. C. H. Huang, H. Ogata, and A. J. Q. Lin, "Predicting Students' academic performance using multiple linear regression and principal component analysis," *J. Inf. Process.*, vol. 26, pp. 170–176, 2018.

[29] L. M. A. Zohair, "Prediction of student's performance by modelling small dataset size," *Int. J. Educ. Technol. Higher Edu.*, vol. 16, no. 1, p. 27, 2019.

[30] Q. Hu and H. Rangwala, "Reliable deep grade prediction with uncertainty estimation," in *Proc. 9th Int. Conf. Learn. Anal. Knowl.*, 2019, pp. 76–85.

[31] Q. Hu and H. Rangwala, "Academic performance estimation with attention-based graph convolutional networks," 2019, *arXiv:2001.00632*. [Online]. Available: http://arxiv.org/abs/2001.00632

[32] K. T. Chui, D. C. L. Fung, M. D. Lytras, and T. M. Lam, "Predicting at-risk university students in a virtual learning environment via a machine learning algorithm," *Comput. Hum. Behav.*, vol. 107, Jun. 2020, Art. no. 105584.

[33] K. T. Chui, R. W. Liu, M. Zhao, and P. O. De Pablos, "Predicting students' performance with school and family tutoring using generative adversarial network-based deep support vector machine," *IEEE Access*, vol. 8, pp. 86745–86752, 2020.

[34] A. Cano and J. D. Leonard, "Interpretable multiview early warning system adapted to underrepresented student populations," *IEEE Trans. Learn. Technol.*, vol. 12, no. 2, pp. 198–211, Apr. 2019.

[35] C. Márquez-Vera, A. Cano, C. Romero, A. Y. M. Noaman, H. M. Fardoun, and S. Ventura, "Early dropout prediction using data mining: A case study with high school students," *Expert Syst.*, vol. 33, no. 1, pp. 107–124, Feb. 2016.

[36] F. Martin and A. Ndoye, "Using learning analytics to assess student learning in online courses," *J. Univ. Teach. Learn. Pract.*, vol. 13, no. 3, p. 7, 2016.

[37] N. Mishra and S. Silakari, "Predictive analytics: A survey, trends, applications, oppurtunities & challenges," *Int. J. Comput. Sci. Inf. Technol.*, vol. 3, no. 3, pp. 4434–4438, 2012.

[38] R. Hasan, S. Palaniappan, S. Mahmood, A. Abbas, K. U. Sarker, and M. U. Sattar, "Predicting student performance in higher educational institutions using video learning analytics and data mining techniques," *Appl. Sci.*, vol. 10, no. 11, p. 3894, Jun. 2020.

[39] F. Widyahastuti and V. U. Tjhin, "Predicting students performance in final examination using linear regression and multilayer perceptron," in *Proc. 10th Int. Conf. Human Syst. Interact. (HSI)*, Jul. 2017, pp. 188–192.

[40] M. H. Abdi, G. Okeyo, and R. W. Mwangi, "Matrix factorization techniques for context-aware collaborative filtering recommender systems: A survey," *Comput. Inf. Sci.*, vol. 2, pp. 8989–8997, Mar. 2018.

[41] Z. Ren, X. Ning, and H. Rangwala, "ALE: Additive latent effect models for grade prediction," in *Proc. SIAM Int. Conf. Data Mining*, 2018, pp. 477–485.

[42] A. Daud, N. R. Aljohani, R. A. Abbasi, M. D. Lytras, F. Abbas, and J. S. Alowibdi, "Predicting student performance using advanced learning analytics," in *Proc. 26th Int. Conf. World Wide Web Companion*, 2017, pp. 415–421.

[43] F. Marbouti, H. A. Diefes-Dux, and K. Madhavan, "Models for early prediction of at-risk students in a course using standards-based grading," *Comput. Edu.*, vol. 103, pp. 1–15, Dec. 2016.

[44] E. Jembere, R. Rawatlal, and A. W. Pillay, "Matrix factorisation for predicting student performance," in *Proc. 7th World Eng. Edu. Forum (WEEF)*, Nov. 2017, pp. 513–518.

[45] I. E. Livieris, K. Drakopoulou, T. A. Mikropoulos, V. Tampakas, and P. Pintelas, "An ensemble-based semi-supervised approach for predicting students' performance," in *Research on e-Learning and ICT in Education*. Cham, Switzerland: Springer, 2018, pp. 25–42.

[46] E. A. Amrieh, T. Hamtini, and I. Aljarah, "Mining educational data to predict student's academic performance using ensemble methods," *Int. J. Database Theory Appl.*, vol. 9, no. 8, pp. 119–136, 2016.

[47] O. W. Adejo and T. Connolly, "Predicting student academic performance using multi-model heterogeneous ensemble approach," *J. Appl. Res. Higher Edu.*, vol. 10, no. 1, pp. 61–75, Feb. 2018.

[48] K. S. Rawat and I. Malhan, "A hybrid classification method based on machine learning classifiers to predict performance in educational data mining," in *Proc. 2nd Int. Conf. Commun., Comput. Netw.* Singapore: Springer, 2019, pp. 677–684.

[49] H. Alaiz-Moretón, J. A. L. Vázquez, H. Quintián, J.-L. Casteleiro-Roca, E. Jove, and J. L. Calvo-Rolle, "Prediction of student performance through an intelligent hybrid model," in *Proc. Int. Conf. Hybrid Artif. Intell. Syst.* Cham, Switzerland: Springer, 2019, pp. 710–721.

[50] G. G. Colombini, I. B. M. de Abreu, and R. Cerri, "A self-organizing map-based method for multi-label classification," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 4291–4298.

[51] N. Saini, S. Saha, and P. Bhattacharyya, "Incorporation of neighborhood concept in enhancing SOM based multi-label classification," in *Proc. Int. Conf. Pattern Recognit. Mach. Intell.* Cham, Switzerland: Springer, 2019, pp. 91–99.

[52] E. Alyahyan and D. Düştegör, "Predicting academic success in higher education: Literature review and best practices," *Int. J. Educ. Technol. Higher Edu.*, vol. 17, no. 1, Dec. 2020.

[53] T. T. York, C. Gibson, and S. Rankin, "Defining and measuring academic success," *Practical Assessment, Res., Eval.*, vol. 20, no. 1, p. 5, 2015.

[54] J. Beel, B. Gipp, S. Langer, and C. Breitinger, "Paper recommender systems: A literature survey," *Int. J. Digit. Libraries*, vol. 17, no. 4, pp. 305–338, 2016.

[55] M. H. Abdi, G. Okeyo, and R. W. Mwangi, "Matrix factorization techniques for context-aware collaborative filtering recommender systems: A survey," Ph.D. dissertation, Dept. Comput. Inf. Sci., De Montfort Univ., Leicester, U.K., 2018.

[56] J. Goldberger, G. E. Hinton, S. T. Roweis, and R. R. Salakhutdinov, "Neighbourhood components analysis," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005, pp. 513–520.

[57] Y. Koren and R. Bell, "Advances in collaborative filtering," in *Recommender Systems Handbook*. Boston, MA, USA: Springer, 2015, pp. 77–118.

[58] X. Luo, M. Zhou, Y. Xia, and Q. Zhu, "An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1273–1284, May 2014.

[59] N. Gillis, "The why and how of nonnegative matrix factorization," *Regularization, Optim., Kernels, Support Vector Mach.*, vol. 12, no. 257, pp. 257–291, 2014.

[60] A. Cichocki and A.-H. Phan, "Fast local algorithms for large scale nonnegative matrix and tensor factorizations," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. E92-A, no. 3, pp. 708–721, 2009.

[61] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Nov. 2011.

[62] M. Schneider, A. Kandel, G. Langholz, and G. Chew, *Fuzzy Expert System Tools*. Hoboken, NJ, USA: Wiley, 1996.

[63] M. Negnevitsky, *Artificial Intelligence: A Guide to Intelligent Systems*, 3rd ed. Saskatoon, SK, Canada: Pearson, May 2011.

[64] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *J. Stat. Softw.*, vol. 33, no. 1, p. 1, 2010.

[65] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Mining multi-label data," in *Data Mining and Knowledge Discovery Handbook*. Boston, MA, USA: Springer, 2010, pp. 667–685.

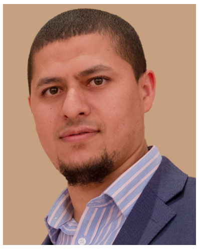
[66] T. Kohonen, "The self-organizing map," *Proc. IEEE*, vol. 78, no. 9, pp. 1464–1480, Sep. 1990.

[67] J. V. G. Tsoumakas and E. Spyromitros. (2020). *MULAN: A Java Library for Multi-Label Learning*. [Online]. Available: http://mulan.sourceforge.net/datasets-mlc.html

[68] G. D. Kuh, S. O. Ikenberry, N. A. Jankowski, T. R. Cain, P. T. Ewell, P. Hutchings, and J. Kinzie, *Using Evidence of Student Learning to Improve Higher Education*. Hoboken, NJ, USA: Wiley, 2015.

[69] A. Namoun, A. Taleb, and M. Benaida, "An expert comparison of accreditation support tools for the undergraduate computing programs," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 9, pp. 371–384, 2018.

ABDULLAH ALSHANQITI received the B.Sc. degree in computer science from Taibah University, Madinah, Saudi Arabia, and the M.Sc. and Ph.D. degrees from the University of Leicester, U.K. He joined the Faculty of Computer Science and Information Technology (FCIS), in 2012, as a Lecturer. He was an Assistant Professor in smart systems and software reverse engineering in 2018. He is currently the Vice Dean of FCIS, Islamic University of Madina, recognized for his work on machine learning, software reverse engineering based on dynamic analysis, model/graph transformations using intelligent learning, and inference approaches. His research interests include research cooperation in different cutting-edge disciplines, including quantum machine learning, hybrid AI approaches that focus on solving NLP, computer vision challenges, and interpretability of deep learning models using graph transformations rules.



ABDALLAH NAMOUN received the bachelor's degree in computer science, in 2004, and the Ph.D. degree in informatics from The University of Manchester, U.K., in 2009.

He is currently an Associate Professor of intelligent interactive systems and the Head of the Information Technology Department, Faculty of Computer and Information Systems, Islamic University of Madinah. He has authored more than 50 publications in research areas spanning intelligent systems, human–computer interaction, software engineering, and technology acceptance and adoption. He has extensive experience in leading complex research projects (worth more than 21 million Euros) with several distinguished SMEs, such as SAP, BT, and ATOS. He has investigated user needs and interaction with modern interactive technologies, design of composite software services, and methods for testing the usability and acceptance of human-interfaces. His recent research interests include integrating state of the art artificial intelligence approaches in the design and development of interactive systems.