Asian Journal of Information and Communications 2020, Vol.12, No. 1, 77-89

KISDI Asian Journal of Information and Communications

Machine learning applications for learning early warning system in Taiwan

Jyh-Jiuan Lin*   Gwei-Hung Tsai**   Ching-Hui Chang***

Tamkang University, Taiwan   Ming Chuan University, Taiwan   Ming Chuan University, Taiwan

Abstract

This research proposes an alternative approach reference for a learning early warning system implementation. Digital e-portfolio data of 6 semesters are used respectively to build 4 commonly used supervised machine learning (ML) classifiers including random forests (RF), support vector machine (SVM), extreme gradient boosting (XGBoost) and artificial neural networks (ANN). The empirical results from year 2013 to 2019 semesters, excluding 2018 due to sabbatical leave of the instructor, show that the top 2 classifiers are XGBoost and RF in terms of the following aggregated criteria consideration: 1. Accuracy, 2. Recall, 3. Precision, 4. *F1-score,* 5. AUC, 6. Cross-validation mean accuracy, 7. Cross-validation accuracy standard deviation (StDev), and 8. Computation time. Since XGBoost has outperformed the rest classifiers, it is recommended to be deployed by the early warning system implementation. The evidence of the model robustness supports the approach of the learning early warning systems implementation incorporating ML methods. Besides, midterm score reaches a consensus for XGBoost and RF to be selected as the most significant features to identify at-risk students. Interestingly, the second most significant feature selected by RF is the “mock exam score”. It fits the purpose of mock exam which is designed to help students foresee the midterm test format. On the other hand, the second most significant feature selected by XGBoost is the “forum post counts” which implies that the higher the participation is, the better the academic performance gets empirically.

Keywords: e-learning, e-portfolio, recommender system, learning management systems

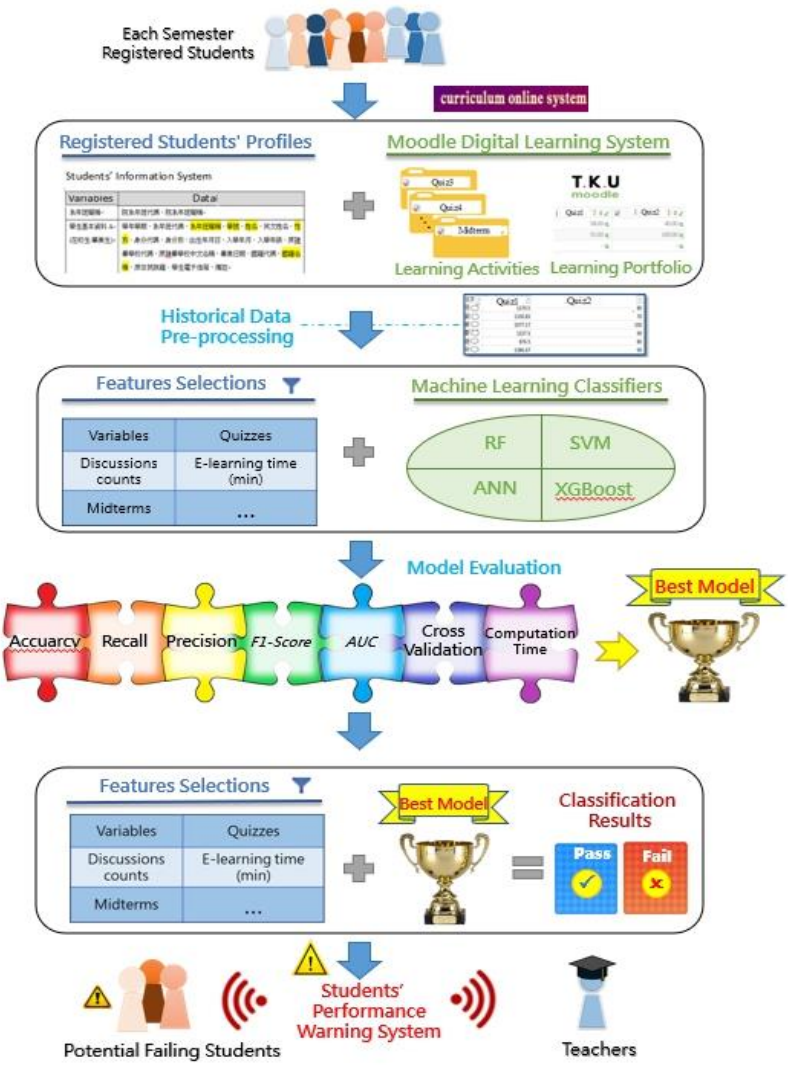
1. Introduction

1.1 Research problems

* 117604@gms.tku.edu.tw
** herbtsai@mail.mcu.edu.tw
*** chchang@mail.mcu.edu.tw
We sincerely thank Hung-Te Chiang for his devotion to the programming. The reviewers’ insightful and helpful comments on our manuscript are highly appreciated.

Students' learning early warning system is highly valued by most schools. The purpose of adapting the early warning system is that it can be used to urge students to devote more time and effort to the coursework. At present, commonly used input variables for implementing the early warning system are related to the attendance, in-class tests grades and participation factors, etc. Other alternative is subject to the teachers' opinion to identify at-risk students. It mainly focuses on the degree of students' participation in the class. The importance of learning early warning system is to remind students as well as teachers in advance and is naturally self-evident. This study eulogizes the practice of the current student learning early warning system, however, this study hopes to provide a different reference for another early warning method for students' digital learning performance by incorporating supervised machine learning (ML) classifiers.

Figure1. Learning early warning system implementation



1.2 Research framework

The issues to be explored in this study are based on the long-distance asynchronous digital learning courses 'Applied Statistical Methods' that conducted from year 2013 to 2019 for 6 semesters in Tamkang Univerity, Taiwan. The research framework of the study is shown in Figure 1.

The students' e-portfolio data from the MOODLE learning platform (including students' academic performance and participation related observations, etc.), along with the students' demographic background data are used to establish four ML classifiers (or models) and classify (the result is used to predict) students to either 'fail' or 'not fail' category according to different models' decisions. In addition, the classification results of 4 ML classifiers are evaluated using 8 different criteria, 1. Accuracy, 2. Recall, 3. Precision, 4. *F1-score*, 5. AUC, 6. Cross-validation mean accuracy, 7. Cross-validation accuracy standard deviation (StDev), and 8. Computation time. Then the best ML classifier is recommended as an alternative reference of learning early warning system for screening at-risk students. The approach of the learning early warning systems implementation in this study is summarized as in the above Figure 1.

2. Machine learning and academic performance classification

2.1 Learning early warning system

Early warning system generally refers to the deployment of intelligent technology systems in nature. Through data collection and analysis, the analysis results are communicated by the system to individuals or groups that may be dangerous in the future. Its main purpose is to allow the recipient of the message to prepare for the imminent danger and take corresponding measures or actions to avoid or mitigate the harm or consequences. On the other hand, the learning early warning system can stimulate students to study harder, maintain students' learning quality, and strengthen students' academic counseling mechanism, so as to maintain students' learning effectiveness. Therefore, most of the early warning systems in the existing colleges and universities are mostly focused mainly on the at-risk students right after midterm examination. Then school administration office will initiate all kinds of remedial strategies through learning tutor guidance, course counseling, even inform parents and other mechanisms to enhance and strengthen student learning effectiveness.

It cannot be emphasized enough that the establishment of a learning early warning system is really essential. By collecting data from students' learning behaviors, with the help of ML classifiers, this study focuses on the selection of classifiers to provide an alternative reference of learning early warning system.

2.2 Machine learning classifiers in academic performance prediction

ML methods use algorithms to construct models to find patterns from a large amount of data, which is an indispensable part of artificial intelligence. As decentralized computing capabilities are maturing, computer computation speeds have also increased significantly, and ML methods have become more widely applied to , for instance, securities market analysis, natural language processing, etc. ML methods can be classified into three categories: supervised learning, semi-supervised learning and unsupervised learning. Since the data contains the student's final grade tag information, this study adopted supervised ML methods to proceed the academic performance analysis.

Although early on, e-learning issues have been foreseen and deployed an effective Learning Management Systems by Govindasamy (2001), it was only in recent years that to cope with ML methods to predict students' academic performance draws lots of attentions.

The input variables adopted by Acharya and Sinha (2014) are gender, income, absence and total score. The ML classifiers used include sequential minimal optimization (SMO) and naïve Bayes classifiers, 1-nearest neighborhood and multi-layer perceptron of ANN with forward structure. The conclusion of the study shows that the SMO algorithm has the best prediction effect on students' academic performance, and the accuracy is 66%.

De Albuquerque et al. (2015) used grades, learning cycles, and school scoring as input variables. The only ML classifier used is ANN. The conclusion of the study shows that the accuracy rate of students' academic performance prediction is 85%.

The input variables used by Marbouti et al. (2016) are scores, absences, quizzes, weekly homework, team participation, project milestones, mathematical model action tasks, and physical course test scores, and the ML classification used includes logistic regression, SVM, decision tree, naïve Bayes and ANN. The study shows that the Naïve Bayesian has the best prediction on students' academic performance and the accuracy rate is 85%.

The input variables used by Liu and d'Aquin (2017) are demographic variables and online learning related variables, and the ML classifier used is the unsupervised ML classifier k-prototypes clustering algorithm. The study concluded that the successful student population mainly comes from powerful families, and most of them will complete their higher education.

The input variables adopted by Kumar and Garg (2018) are continuous variables related to school learning assessments. The ML classifiers used are generalized linear model (GLM), multi-layer perceptron (MLP), and gradient boost model, random forests, and deep neural networks. The research concludes with each method to predict students' the academic performance.

3. Research methods

3.1 Supervised machine learning classifiers

The difference between supervised learning and unsupervised learning is whether data is labeled. Supervised learning will first label data, and the classifier will use it for model training. For example, to determine whether a student fails, label the students who fail as "1", and the students who pass as "0" in advance. Supervised learning methods are adopted in this research since our data is labeled. Four following ML classifiers are adopted in this study.

3.1.1 Random forests

To overcome the issue that decision trees are prone to overfitting (Fan, 2013), Breiman (2001) proposed a random forest classifier whose idea originated from the decision tree classifier. Decision tree classifier is a method of classifying a large amount of training data. Each segmentation divides the existing data into two, and then uses the threshold to determine the segmentation. If the data is greater than the threshold, it will be divided to the right, and if it is smaller than the threshold, it is split to the left. When the training data comes to the node, it is determined by the corresponding information gain of the data, whether or not to split the child nodes. In addition to the common information gain (IG), entropy and Gini impurity and mean decrease are other alternatives to evaluate the amount of information. The higher the IG gets or impurity mean decreases, the better the classifiers are. RF classifier, first of all, generates many decision trees and each tree will grow completely without pruning. The classification result of the RF classifier is determined by the voting of each decision tree and used as the final output result, that is, the majority decides the final classification.

3.1.2 Support vector machine

Support vector machine (SVM) proposed by Cortes and Vapnik (1995) can be used for classification and regression in both linear and nonlinear ways. The principle of SVM is that it first converts the original data into a higher dimension space. From these dimensions, some features can be used in the training data set to find the corresponding hyperplane to segment data, and these sample points closest to the boundary are called support vectors because they provide the most classification information. For non-linear problems, Boser et al. (1992) use the kernel function to convert the non-linear binary classification data into a divisible linear space. In this way, it is easier to find a super plane to separate different types of data. Kernel functions are commonly including linear, polynomial, radial basis function, sigmoid, etc.

3.1.3 Extreme gradient boosting

Ensemble learning is a popular method to improve the accuracy and error rate of a single ML classifier. Ensemble learning includes bagging, boosting or stacking three ways. Breiman (1997) proposed that gradient boosting decision tree is a part of boosting algorithm. The spirit of boosting is that as old Chinese saying "Three heads are better than Liang Zhuge.". It is a kind of ML technology used for weighted majority voting classification that connects several weak models in series. It can reduce the deviations. In short, it is to integrate several weak classifiers into a stronger classifier. Chen and Guestrin (2016) extends the gradient boosting decision tree to propose an improved extreme gradient boosting algorithm referred to as XGboost. It still retains the spirit of ensemble learning. Assembling multiple weak classifiers can turn into a stronger classifier and is currently one of the algorithms that stands out in the Kaggle competition.

3.1.4 Artificial neural network

Artificial neural network (ANN) is inspired by the research of neuron in biology and was invented in 1958 by psychologist Frank Rosenblatt, which was pointed out by Alexx Kay in 2001.[1] The nervous system is composed of multiple neurons. Lots of later applications of ANN intend to study human cognition. The basic structure of ANN can be disassembled into: input layer, hidden layer and output layer. The input layer is mainly used to receive the input information. The output layer outputs the processed message. The most important key layer is the hidden layer. The hidden layer is located between the input layer and the output layer. It mainly deals with the interaction of the input layer neurons. The quantity of hidden layers does not have certain specifications and standards. It is found by trial and error. Through the input of the basic unit, neuron, the output of other neurons are endowed with the weights. The larger the weight is, the stronger the connection is, the greater the impact of neurons on the neural-like network becomes; conversely, the smaller the weight is, the less the impact of the connected neuron on the neural-like network bocomes. Therefore the connected neurons will be removed to save time and space if the corresponding weight is too small. Before the neural network is trained, its output is messy. However, as the number of training increases, the neural network's binding value will be gradually adjusted to the error convergence between the target value and the output of the neural network. Usually we will define a cost function as the neural network convergence indicators: as the number of network trainings increases, the cost function will decrease and eventually

[1] https://www.computerworld.com/article/2591759/artificial-neural-networks.html

converge. Therefore, proper learning times can lead to a better performance of neural network classifier. In addition, because there are too many complicated conversion processes of activation functions in the hidden layer, it is difficult to explain the relationship between the input information and the output result, so being not easy to interpret the result is the disadvantage of ANN.

3.2 Classifiers evaluating criteria

After the ML classification model is built, it still needs to be evaluated to see whether the model is appropriate. This paper adopts the aforementioned eight criteria as well as the computational time to evaluate our classification models. In the ML methodology, the actual status of the issue and the classification results of the model are often made into a confusion matrix to facilitate further analysis of the model. This paper focuses on students who actually fail the course. Therefore, students who fail the course will be regarded as «positive» and students who does not fail the course will be regarded as «negative». It leads us to four situations (i) when the students who fail actually and are classified by the model to be students who would fail are referred to as «true positive». (ii) Students who actually fail are classified to be students who would not fail are referred to as «false negative». (iii) Students who actually do not fail are classified to be students who would not fail will be regarded as «true negative». (iv) Students who actually do not fail are classified to be students who would fail will be regarded as «false positive». It can be summarized into the confusion matrix as in Table 1.

Table1. Confusion matrix

classification / actual situation	fail	not fail
fail	true positive(TP)	false negative (FN)
not fail	false positive(FP)	true negative(TN)

3.2.1 Accuracy

Prediction accuracy is defined as in equation (3.1). The accuracy rate is used to measure the accuracy of the model's classification results, however, the accuracy rate might be biased when the data is imbalanced.

$$Accuacy\ = (TP + TN)/(TP + FN + FP + TN) \tag{3.1}$$

3.2.2 Recall

Recall is defined as in equation (3.2) and is also called sensitivity. It indicates the proportion of students who actually fail and are classified to be the 'fail' category. The higher the recall rate is, the better the classification model gets in identifying the students who actually fail.

$$Recall = TP/(TP + FN) \tag{3.2}$$

3.2.3 Precision

Precision is defined as in equation (3.3) and indicates the proportion of students classified to be the 'fail' category who actually fail. The higher the precision is, the less likely the model is to misclassify students who actually do not fail.

$$Precision\ Rate = TP/(TP + FP) \tag{3.3}$$

3.2.4 F1-score

*F*1-*score* is defined as in equation (3.4). It can be seen that *F1-score* is defined as the harmonic mean of recall and precision rate. *F1-score* is more sensitive to extreme values. The higher the value is, the more robust the classification model becomes.

$$F1 - score = 2/(1/Precision + 1/Recall\ ) \ = \ 2TP\ /(2TP + FP + FN) \tag{3.4}$$

3.2.5 AUC

AUC is the area under the Receiver Operating Characteristic (*ROC*) curve and above the coordinate axis and is defined as in formula (3.5). *AUC* is an indicator for model performance. The larger the *AUC* value, the better the model performance.

$$AUC = \int_0^1 TP(FP)d(FP) \tag{3.5}$$

Fawcett (2006) once mentioned that *AUC* is a part of the unit square area (side length 1 unit). Therefore, *AUC* must be between 0 and 1. *AUC* of 1 means this is a perfect scoring model; in addition, when the *ROC* curve happens to be on the diagonal, it means that its *AUC* is 0.5, which can be regarded as a random model. Echo Hosmer and Lemeshow's (2000) argument for *AUC*, (i) when *AUC* = 0.5, it can be regarded as a random model, the classifier has no discriminatory ability, (ii) when 0.7 $\leqq$ *AUC* < 0.8, the classifier discriminatory ability is acceptable, (iii) when 0.8 $\leqq$ *AUC* <0.9, the classifier has excellent discriminatory ability, (iv) when 0.9 $\leqq$ *AUC*, the classifier has super excellent discriminatory ability.

3.2.6 Cross validation

Cross validation (CV) is a criterion used to evaluate and verify the performance of classifiers. The usual approach is to split the original data into training data sets and test data sets. Training data is used to construct the classifier, and then the test data is used to test the training model, and the classification accuracy is evaluated. CV is used as an index to evaluate the performance of the classifier. CV methods have different approach including (1) Hold-out method, (2) K-fold cross validation method and (3) Leave-one-validation method. 10-fold cross validation method is adopted in this study. Besides the average accuracy of 10 folds, standard deviation of the accuracy is also provided in the empirical result.

4. Empirical results

4.1 Data and input variables

There are two main data sources from Tamkang University in this study. One is the demographic data from the student information system maintained by the Academic Affairs Office and the other is the e-portfolio data from MOODLE digital learning archives maintained by the Long Distance Learning Promotion Center. The input variables used in this study are as follows.

4.1.1 Demographic data

This study uses the input variables of the students' demographic data, including department ID, class ID, student ID, gender, nationality, zip code, school year semester, subject code, moral conduct assessment, award, credits accumulated, grades, class ranking, etc.

4.1.2 MOODLE e-portfolio data

This study uses the input variables of the MOODLE platform digital learning e-portfolio data. It tracks two aspects of students' performance of learning activities. The first aspect is regarding the students' participation. It observes the number of logins, the absence of the exam, the time (minutes) devoted on the platform including viewing the video and other learning activities, the number of post on the discuss forum, etc. before the mid-term exam. The second aspect is regarding the academic performance, the online tests scores, including midterm exams, final test, quizzes before the midterm test, and semester grade. A total of 6 quizzes and 1 mock exam are carried out before the midterm exam for each course studied. In addition, semester grades are also tracked with the purpose of labeling students who fails or not.

4.1.3 Training data and test data

There are overall 996 students enrolled in the courses. Details for each semester and the status are summarized in Table 2. Generally speaking, the failure rate is around 30%. 90% of each individual semester data is randomly selected as the training data to build the classifiers, and the rest 10% data is as the testing data to assess the classifiers' performance. It also combines 6 semester data to build an overall classifiers for the sake of comparison.

Table 2. Data size and status

Semester / Status	2013	2014	2015	2016	2017	2019	Overall
Fail	67	41	54	79	41	20	302
Not Fail	155	89	96	109	124	121	694
Total	222	130	150	188	165	141	996

4.2 Classifier selection

After injecting the training data to the model, it was tested. For the semesters overall from year 2013 to 2019, excluding 2018 (since it is not offered due to sabbatical leave of the instructor), classification performance is evaluated by the eight criteria in Table 3 and table 4. The best models selected by each criterion are counted and summarized in Table 4.

Table 3. Individual year ML classifiers performance measures

Classifier / Criteria	2013				2014			
	RF	SVM	XGboost	ANN	RF	SVM	XGboost	ANN
Accuracy	1.0000*	0.7727	1.0000*	0.8636	1.0000*	0.7692	1.0000*	0.9231
Recall	1.0000*	0.2857	1.0000*	0.5714	1.0000*	0.2500	1.0000*	0.7500
Precision	1.0000*	1.0000*	1.0000*	1.0000*	1.0000*	1.0000*	1.0000*	1.0000*

F1-score	1.0000*	0.4444	1.0000*	0.7273	1.0000*	0.4000	1.0000*	0.8571
AUC	1.0000*	0.6429	1.0000*	0.7857	1.0000*	0.6250	1.0000*	0.8750
CV mean accuracy	0.7500	1.0000*	1.0000*	0.9100	0.7500	1.0000*	0.9583	0.8341
CV accuracy StDev	0.0000*	0.0000*	0.0000*	0.0568	0.0000*	0.0000*	0.0589	0.0801
Comput.time(sec)	0.03	0.02*	0.40	21.61	0.03	0.01*	0.23	19.13
Classifier / Criteria	2015				2016			
	RF	SVM	XGboost	ANN	RF	SVM	XGboost	ANN
Accuracy	0.8667*	0.8000	0.8667*	0.8000	0.8947	0.7895	0.9474*	0.8947
Recall	0.8000*	0.4000	0.8000*	0.6000	0.7500	0.5000	0.8750*	0.7500
Precision	0.8000	1.0000*	0.8000*	0.7500	1.0000*	1.0000*	1.0000*	1.0000*
F1-score	0.8000*	0.5714	0.8000*	0.6667	0.8571	0.6667	0.9333*	0.8571
AUC	0.8500*	0.7000	0.8500*	0.7500	0.8750	0.7500	0.9375*	0.8750
CV mean accuracy	0.7500	1.0000*	0.9709	0.9121	0.7500	1.0000*	1.0000*	0.8958
CV accuracy StDev	0.0000*	0.0000*	0.0376	0.0605	0.0000*	0.0000*	0.0000*	0.0724
Comput.time(sec)	0.03	0.02*	0.4	19.95	0.03	0.02*	0.52	21.02
Classifier / Criteria	2017				2019			
	RF	SVM	XGboost	ANN	RF	SVM	XGboost	ANN
Accuracy	0.8750*	0.7500	0.8750*	0.8750*	1.0000*	0.8571	1.0000*	0.8571
Recall	0.7500*	0.0000	0.7500*	0.5000	1.0000*	0.0000	1.0000*	0.0000
Precision	0.7500	0.0000	0.7500	1.0000*	1.0000*	0.0000	1.0000*	0.0000
F1-score	0.7500*	0.0000	0.7500*	0.6667	1.0000*	0.0000	1.0000*	0.0000
AUC	0.8333*	0.5000	0.8333*	0.7500	1.0000*	0.0000	1.0000*	0.0000
CV mean accuracy	0.7500	1.0000*	1.0000*	0.8677	0.8571	0.859	1.0000*	0.8603
CV accuracy StDev	0.0000*	0.0000*	0.0000*	0.0674	0.0000*	0.0307	0.0000*	0.0297
Comput.time(sec)	0.03	0.01*	0.69	20.58	0.04	0.02*	0.42	21.56

* criterion optimal performance in each year

Table 4. ML classifiers performance measures and best model counts

Classifier / Criteria	Overall(2013~2019) **				Best model counts				
	RF	SVM	XGBoost	ANN	Year	RF	SVM	XGBoost	ANN
Accuracy	0.9293	0.8283	0.9394*	0.9394	2013	6	4	7	1
Recall	0.9000	0.5000	0.9333*	0.8677	2014	6	4	5	1
Precision	0.8710	0.8824	0.8750	0.9286*	2015	5	4	5	0

F1-score	0.8852	0.6383	0.9032*	0.8966	2016	2	4	7	1
AUC	0.9210	0.7355	0.9377*	0.9188	2017	5	3	6	2
CV mean accuracy	0.8462	1.000*	0.9598	0.8719	2019	6	1	7	0
CV accuracy StDev	0.0000*	0.000*	0.0192	0.0257	overall*	2	2	4	1
Comput. time(sec)	0.09*	0.17	1.27	59.83	Average	4.57	3.14	5.86*	0.86

* criterion optimal performance in each year, **2018 is excluded

4.3 Feature importance

After the early warning system has identified the at-risk students, the most highly related features information could help teacher to take further effective remedial guidance to the at-risk students. The top 5 ranked features from each semester and the total come to 12 variables including scores of 6 quizzes (Quiz1~Quiz6), mock exam, midterm, and browsing time on the platform (browsing), post counts on the in-class forum (forum post #), major, and total credits obtained are listed. The top 5 ranked (Rk) feature importance from each year incorporating the impurity mean decrease (Impurity) measure and information gain(Gain) are summarized in Table 5 and Table 6 for RF (the second best) and XGBoost (the best) classifiers, respectively.

Table 5. Top 5 feature importance in RF classifier

year	2013		2014		2015		2016		2017		2019		average
Performance / Features	Rk	Impurity	Rk	Impurity	Rk	Impurity	Rk	Impurity	Rk	Impurity	Rk	Impurity	Impurity
Quiz1	-	-	-	-	-	-	-	-	-	-	2	5.370	0.895
Quiz2	5	4.640	-	-	-	-	-	-	-	-	5	2.840	1.247
Quiz3	-	-	-	-	-	-	5	3.220	4	5.54	3	4.060	2.137
Quiz4	-	-	4	4.990	-	-	2	16.08	2	9.08	-	-	5.025
Quiz5	-	-	-	-	2	5.430	-	-	-	-	-	-	0.905
Quiz6	2	11.37	-	-	5	4.880	-	-	5	3.860	-	-	3.352
mock exam	3	9.440	1	10.94	4	4.960	3	7.160	1	11.52	4	3.750	7.962
midterm	1	33.01	3	5.770	1	7.440	1	32.30	3	7.260	1	6.480	15.377*
browsing	-	-	5	3.320	-	-	4	3.440	-	-	-	-	1.127
forum post#	-	-	2	6.890	3	5.160	-	-	-	-	-	-	2.008
major	4	5.480	-	-	-	-	-	-	-	-	-	-	0.913
credits	-	-	-	-	-	-	-	-	-	-	-	-	0

* maximum

Table 6. Top 5 feature importance in XGboost classifier

year	2013		2014		2015		2016		2017		2019		average
Performance / Features	Rk	Gain	Rk	Gain	Rk	Gain	Rk	Gain	Rk	Gain	Rk	Gain	Gain
Quiz1	2	0.149	-	-	-	-	5	0.091	-	-	-	-	0.040
Quiz2	-	-	-	-	-	-	3	0.100	-	-	-	-	0.017
Quiz3	-	-	-	-	2	0.174	4	0.093	-	-	5	0.080	0.058
Quiz4	-	-	-	-	-	-	-	-	-	-	-	-	0
Quiz5	-	-	-	-	4	0.107	-	-	4	0.097	1	0.252	0.076
Quiz6	-	-	4	0.095	5	0.099	-	-	3	0.140	-	-	0.056
mock exam	-	-	-	-	-	-	-	-	-	-	-	-	0
midterm	3	0.142	2	0.152	3	0.121	1	0.289	1	0.166	2	0.167	0.173*
browsing	4	0.111	5	0.094	-	-	-	-	-	-	-	-	0.034
forum post#	1	0.151	3	0.139	1	0.217	2	0.201	2	0.152	3	0.125	0.164
major	-	-	-	-	-	-	-	-	-	-	-	-	0
credits	5	0.098	1	0.159	-	-	-	-	5	0.093	4	0.099	0.075

* maximum

Speaking in average, if one uses the second best classifier RF, “midterm score” and the “mock exam score” are the most (Impurity measure is 15.377) and the second most (Impurity measure is 7.962) significant feature to interpreting the classification results in terms of impurity mean decrease measure. On the other hand, if one uses the best classifier XGBoost, “midterm scores” and “forum post counts” are the most (Gain measure is 0.173) and the second most (Gain measure is 0.164) significant feature to interpreting the classification results in terms of information gain measure.

Even though the features selection from the two classifiers are not exactly the same, midterm score is the consensus choice. It means possibly that at-risk students may have learning difficulties in this subject or simply personal neglecting or other evitable excuses like lousy time management. Interestingly, the second most significant feature selected by RF is the “mock exam score”. It fits the purpose of mock exam designed for helping students to foresee the midterm test format. The empirical result implies it. On the other hand, the second most significant feature selected by XGBoost is the “forum post counts” which shows students’ class participation degree.

5. Conclusions

From the students’ academic performance aspects, the empirical evidence from 2013 to 2019 semesters, excluding 2018 shows that midterm score is the consensus on the most significant features to identify at-risk students. Interestingly, the second most significant feature selected by RF is the “mock exam score”. It fits the purpose of mock exam designed for helping students to foresee the midterm test format. On the other hand, the second most significant feature selected by XGBoost is the "forum post counts" which indicates students' class participation degree. Therefore, the empirical results implies that students could have better academic performance with extra help (mock exam) and more devotion.

On the ML aspects, the empirical evidence shows the best classifier is XGBoost in our study in terms of the following aggregated criteria consideration: 1. Accuracy, 2. Recall, 3. Precision, 4. *F1-score*, 5. AUC, 6. Cross-validation mean accuracy, 7. Cross-validation accuracy standard deviation, and 8. Computation time. XGBoost has outperform the rest classifiers and therefore is recommended to assist the learning early warning system implementation for future learning early warning system implementation references. The evidence of the model robustness supports the approach of the learning early warning systems implementation. Therefore, this study should also be extended and applied easily to other courses for identifying at-risk students once they have kept performance records of the students. However, model robustness could vary due to some other unspecific factors and should be validated through long term historical data establishment.

References

Acharya, A. and Sinha, D. (2014). Early prediction of students performance using machine learning techniques. *International Journal of Computer Applications*, *107*(1), 37–43.

Boser, B. E., Guyon, I. M. and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. *COLT '92 Proceedings of the fifth Annual Workshop on Computational Learning Theory*, 144-152.

Breiman, L. (1997). Arcing the edge. *Technical Report 486*, Statistics Department, University of California, Berkeley.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5-32.

Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, California, USA, 785-794, ACM.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297.

De Albuquerque, R. M., Bezerra, A. A., De Souza, D. A., Do Nascimento, L. B. P., De Mesquita Sá, J. J. and Do Nascimento, J. C. (2015). Using neural networks to predict the future performance of students. *IEEE International Symposium on Computers in Education* (*SIIE*), 109–113.

Fan, H. (2013). Land-cover mapping in the Nujiang Grand Canyon: integrating spectral, textural, and topographic data in a random forest classifier. *International Journal of Remote Sensing*, *34*(21), 7545-7567.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861-874.

Govindasamy, T. (2001). Successful implementation of e-learning: pedagogical considerations. *The Internet and Higher Education*, *4*(3), 287-299.

Hosmer, D. W. and Lemeshow, S. (2000). *Applied Logistic Regression, 2nd ed.* New York, Chichester, Wiley.

Kumar, V. and Garg, M. L. (2018). Comparison of machine learning models in student result prediction. *International Conference on Advanced Computing Networking and Informatics*, 439–452.

Liu, S. and d'Aquin, M. (2017). Unsupervised learning for understanding student achievement in a distance learning setting. *IEEE Global Engineering Education Conference*, 25–28.

Marbouti, F., Diefes-Dux, H. A. and Madhavan, K. (2016). Models for early prediction of at-risk students in a course using standards-based grading. *Computers and Education*, *103*(1), 1–15.