DETECT FAIL STUDENTS USING MACHINE LEARNING DURING AN E-LEARNING COURSE ASSESSMENTS

[1]Mushtaq Hussain, [2]Wenhao Zhu, [3] Wu Zhang, [4]Syed Muhammad Raza Abidi, [5]Hu Guannan
School of Computer Engineering and Science, Shanghai University, 99 Shangda Road, Baoshan District, Post Code 200444: Shanghai, China.
Contact: whzhu@shu.edu.cn

ABSTRACT: *There are a number of problem-related with e-learning systems. The most important of these problems is forecasting fail student during the course assessments. In this project, we used machine learning (ML) algorithms to predict the fail student's during course assessments of an open university (OU) in virtual learning environment (VLE) in which students must solve distinct course assessments. The machine learning algorithms (i.e., Generalized linear model (GLM), Deep learning (Dl), Random forest (RF) and Gradient Boosted Trees (GBT) classifiers) were used in current study. Using these algorithms, first all proposed ML classifiers were trained and then the accuracy of the models were obtained on unknown data. The results show that the RF and DL classifiers offer higher performances than the other models. RF and DL classifiers can easily be integrated into VLE systems to assist a teacher to monitor the performance of students during a VLE course and provide intervention for those students in advance of the final exam.*

Keywords: Machine learning, Student performance, virtual learning environment, fail student

INTRODUCTION

E-learning has the excessive contribution to higher education. The massive open online courses (MOOCs), intelligent tutoring systems and online universities are the major types of e-learning systems. It enables the student to study from any place [16].

Currently, computer is used in many domains such as e-learning, recommendation, pattern recognition, image processing, medical diagnosis, and many others [11].

Evaluating student performance and predicting fail students are two important problems for online education and traditional education. Teachers find difficulty to evaluate the performances of individual students in e-learning systems [11]. Traditional universities have used tests to evaluate student performance and have spent enormous amounts of time developing standardized tests [7].

ML techniques can function as appropriate tools for evaluating student performance and, by calculating the probabilities of fail students in a course, can identify fail students during learning sessions. Using ML tools, teachers can recognize these fail students at an early stage before the course is complete. Additionally, ML techniques can help an e-learning teacher find an appropriate difficulty level for a group of students and allow the teacher to prepare class lectures accordingly. Using an ML algorithm, the teacher can also alert fail students at the early stages of a course [11]. Sometimes it is hard to choose best ML algorithms for solving real world problems because their prediction performance depends on their parameters, features and the problem domain. Researchers have used various methods and features in predicting student performance. Mostly, they have applied students' demographic information such as grades, age, marital status, number of children, and occupation as features to predict student performance when training supervised ML algorithms [11]. However, many of these features are not easily available to researchers; they must expend considerable effort to select these features from the raw data. Feature selection can affect the precision and accuracy of ML algorithm prediction models. Moreover, most studies have focused on predicting student performance at the end of the course or session; they were not intended to identify fail students in a timely manner.

In the current study, we used score of the student in different assessment instead of demographic data to predict student performance during an e-learning system session. score features are readily available and were used in this study to determine which course were particularly difficult for students. We used several different ML algorithms to predict fail student performance during a virtual learning environment (VLE) session of open university (OU) and then compared the algorithms' performances. Our models can integrate smoothly into e-learning systems and enable teachers to identify fail students. When classifier detects some course difficult then instructor gives extra time to the course and provides extra material to students. Overall, ML models can help minimize the percentage of students who fail final exams.

This study involved the following steps:

- Build the ML models for the generalized linear model (GLM), deep learning (DL), random forest (RF) and gradient boosted trees (GBT) classifiers;
- Adopt a score of student to predict fail student in VLE course assessments; and
- Evaluate the model results to identify the best model for predicting fail students.

Related work is discussed in Section 2, and the proposed techniques are presented in Section 3. Section 4 describes and discusses the experimental results. Finally, Section 5 provides conclusions and outlines future work.

RELATED WORK

Identifying fail students in VLE course assessments is important because once identified, teachers can intervene to provide help at an early stage (i.e., before the final exam). Many researchers have been conducted to predict student performance. These studies have utilized various features such as student history and age, as well as other demographic information such as profession and chosen domain. The researcher also used different ML techniques such as SVM [6, 9], ANN to predict student performance.

However, to our knowledge, no research has been conducted to predict student performance in a web-based education (such as VLE) using grade of student. Kotsiantis et al. [11] predicted student performance with ML algorithms using students' assignment marks to predict performance and then compared those with alternative ML algorithms. Researchers have demonstrated that the NB classifier achieves good accuracy compared to other classifiers. Vahat *et al.* [18] used a simulated DEEDS dataset to study students' learning behavior utilizing analytics. They compared student groups with their academic exam results and showed that students' grades depend more on their learning behaviors than on the difficulty of the material. Acharya and Sinha [1] predicted student performance at an early stage using features such as gender, socio-economic status, religion, and family size with classifiers that included decision trees, Bayesian networks, ANNs, and SVMs. Arora et al. [3] used the radial basis function (RBF) to predict student grades for a semester. Acikkar et al. [2] used an SVM to predict whether students would be admitted to a school. Researchers have also used physical test records to predict student performance. Sharma et al. [15] applied an LR model to predict student placements using secondary and higher secondary school test grades. Zheng et al. [19] adopted neural networks to predict students' grades using features than included VOD time; courseware download times, BBS posting time, and assignment submission time.

All the above work used traditional features of students to predict performance in different domains. In some situations, having numerous features or large amounts of training data does not increase model accuracy.

PROPOSED METHODS

The goal of this study is to predict fail students in a VLE using ML algorithms. As the data were not ready for apply ML algorithms therefore first we apply preprocessing steps, we used GLM, DL, RF and GBT ML algorithms to build the learning models. These classifiers are supervised learning algorithms, therefore, these classifiers first trained on training data and then tested on previously unseen data. This study utilized a student score during assessment to predict fail student with all the tested classifiers. Then, we compared the classification results and, finally, identified the best model for our data set.

In this study, students were given five assessments to solve during course. Individual students spent varying amounts of time on each assessment. We utilized five features to predict fail students' and determine whether their command of the information exhibited any weaknesses during the VLE session. The input score feature is shown in Table 1. Cross-validation process was used to compare the performance of current study classifiers [14]. The details of the proposed models are described below.

Generalized Linear Model (GLM)

The GLM model can be used as classification technique whose output is between 1 and 0. The GLM model was applied to predict the fail students using the relationship between a student's final exam results and the score that student achieved completing the assessments. The details of this model are given below.

It is supervised learning model which is used for both regression and classification problem. GLM are the generalized shape of linear regression, ANOVA, Poisson regression etc [10].

We used the default parameters value of RapidMiner to build GLM and then estimated the correct model results, as discussed in Section 4.

Deep Learning

Deep learning based on multi-layer feed-forward artificial neural network. ANNs learn from training data and are then tested with unseen data [8,12]. During DL process, a back-propagation algorithm and stochastic gradient descent is utilized to compute the theta value to obtain the desired output value [13]. DL model contains large number of hidden layers and neurons. DL typically performs well when the numbers of features are large. The input nodes accept input data. The layers between the input and output layers are called "hidden layers." The hidden layers perform operations on the input data and pass the results to other neurons, called "output nodes." Hidden nodes are also called activation nodes or node values.

An DL can use a tanh function and maxout activation function in the hidden layer to compute a value that is passed to another hidden layer [4]. Due to using advance features (adaptive learning rate, rate ennealing, momentum training) increased the model accuracy.

Random Forest (RF)

The RF is a popular classifier for classification and regression problems. It is simple ML algorithms and gives good result without parameter tuning. RF model combined certain number of decision trees and gives more accurate and stable prediction. These trees are trained with training data, and then obtained a high accuracy on the test data. In RF Model, the relative importance between features in prediction is easily measured. Furthermore, RF has many decision trees; therefore, it prevents over fitting. The disadvantage of the RF is that, due to large number of decision trees, it slows down the performance in real life prediction.

The subset ration criterion is used for splitting rule selection. Section 4 discusses the results of the RF model.

Gradient Boosted Trees (GBT)

The GBT classifier is used for regression and classification problem. It is the ensemble of weak models and reduce the over fitting problem by using regularization method. The GBT is nonlinear in nature, therefore it produce better accuracy. Additionally, it uses loss function for optimization.

Here, the predictor variable is the score of students achieved completing different assessments, as shown in Table 1.

We first trained the GBT model with the training data and then tested it with unknown data. Finally, we obtained an F1-score and calculated the precision of the model. These results are shown in Section 4.

EXPERIMENTS AND RESULTS

In this project, we predicted student performance by training different ML classifiers using student score (the score of students, after completing several different assessments). We used ML algorithms and the Rapid minor tool to build the learning models as described below.

Data Description

As the data for this study, we utilized the Open University learning analytics dataset [20]. We studied the data from Open University students working on the VLE system, which focuses a particular topic in a given assessment. This OU provides topic-related materials to students through a VLE. This virtual learning environment (VLE) delivered several assessments to social science course students, who each spent varying amounts of time trying to solve each assessment. The final result and the score each student spent on each assessment are all included in the data for this course [20].

ML algorithms learn from applying patterns to data. The raw data of current study was not ready to apply the ML, therefore, we performed some preprocessing steps using Microsoft Excel 2013. These preprocessing steps are listed below.

- Datasets with many attributes have some disadvantages: they are computationally expensive, exacerbate data overfitting problems, and reduce the generalization ability [1]. Therefore, we removed all the unwanted columns to select only a subset of features (columns) from the raw data. These features are the score obtained on each assessment. We utilized these features to predict student's performance by removing the other features from the treated data.
- We prepared data for a machine learning algorithm, where each row index presents a student ID and each column index is the score in assessment. The attributes extracted from the raw data are listed in Table 1, in which the columns represent the model features and the rows are the students' records. Thus, each attribute represents the score obtained by a student to solve an assessment.
- We saved our data set in an .xlsx format in one page: a feature page (X). In the feature page, the rows represent the score obtained on each of the five assessments. The label column contains the students' final result in the final exam.
- The current study dataset contains missing records. We replaced zero for these missing records.
- In the second step, we standardized the data by applying the z-score feature-scaling technique to adjust all data values between 0 and 1. Our dataset contained five attributes and two classes: 0 denotes fail student final exam and 1 designates pass student in the final exam.

Table 1. Datasets of current study

Ass1_Score	Ass2_Score	Ass3_Score	Ass4_Score	Ass5_Score
a_{11}	a_{12}	a_{13}	a_{14}	a_{15}
a_{21}	a_{22}	a_{23}	a_{24}	a_{25}
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.

Note: Assessment score (Ass_Score), all above score shows the score obtained in five assessment

RESULTS AND DISCUSSION

The study was investigated that which of the machine learning algorithms are most appropriate for predicting fail students from their performances during a VLE course. We performed experiments using two performance measures: cross-validation and the receiver operator characteristic (ROC) curve. We used Cross-validation to check model performance, and there are two types: k-fold cross-validation and leave-one-out cross-validation. According to the k-fold cross-validation process, first, the total data of current study were divided into k different portion. Second, the model of current study is trained from k-1 subsets. Finally, the remaining subset is used to check the classifier performance on testing data. This performance received from n-fold cross validation will be considered a good performance guess for the model. The model obtained from this process will be considered a generalized model for a whole dataset [5].

During the experiment, the input features were the score' students obtained on the different assessment and the target variables were the students' final exam result. We divided the data using a cross-validation method to evaluate the performance of each model. In this experiment, the sizes of the test and training samples were 20% and 80%, respectively.

We used generalized linear model (GLM) with default parameters and evaluated GLM model with the cross-validation process. For the cross-validation, we divided the data into two components: a training data set and a test dataset. We trained the GLM model using default parameter

We obtained different accuracy as shown in Figure 1, Table 2.

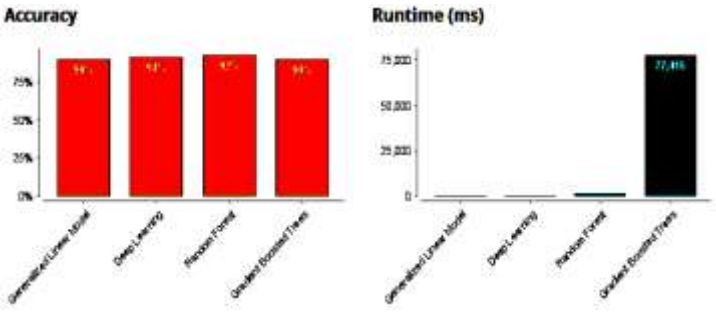

Fig 1. Accuracy of current study models'.

Next, we evaluated the Deep Learning models using the cross-validation method. We investigated with default parameter values in RapidMiner. Finally, our model yielded the best performance when using default parameter. We used the RapidMiner tool to obtain the performance results shown in Table 2.

We used a RapidMiner tool to train the random forests model (RF) and obtained the highest accuracy using the default parameters, as shown in Table 2. The best accuracy achieved by the RF was 92%.

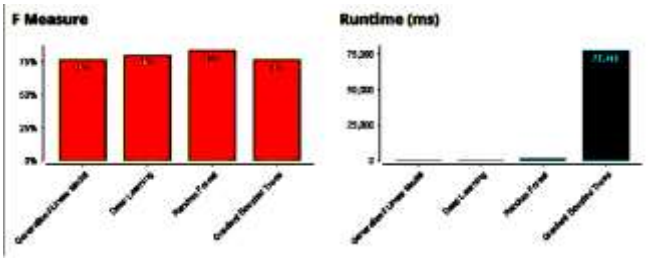

Fig 2. F measure of current study models'

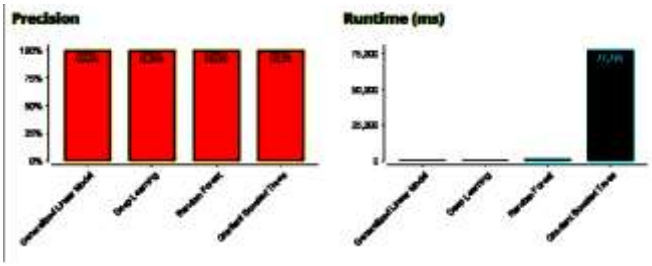

Fig 3. Precision of current study models'

Finally, we trained the Gradient Boosted Trees (GBT) classifier to our training data to find the of fail and pass students. We checked the GBT performance with default parameter of RapidMiner and ultimately achieved a good performance using a test data. The results are listed in Table 2. The accuracy of the GBT classifier was 90%.

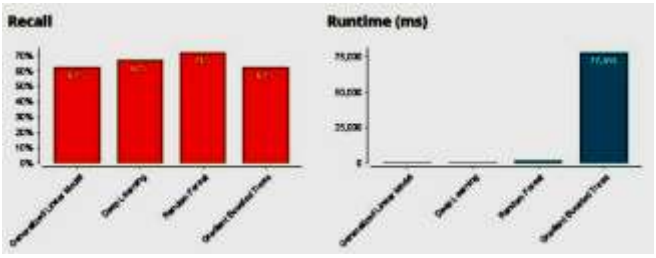

Fig 4. Recall of current study models'

After producing the learning models, we investigated how precise these models were. We assessed our models using several different measures. The first is accuracy, which finds the fail students that the models correctly predicted to be fail. We calculate the prediction results for all the current study models and calculate the true positives, true negatives, false positives, and false negatives. We also determined precision scores for all the models from these values. High precision values indicate that the probability of the test set being accurately classified will be high.

In this study, we want to predict fail students in a social science course session, therefore precision shows the fractions among them who truly fail students were:

Precision = True Positive/True Positive + False Positive

Next, we computed the recall measure, which finds the ratio of all the fail students in the data set who truly do not achieved high grade that the classifiers exactly recognized as fail.

Recall = True Positive/True Positive + False Negative

If model have high recall scores then its performance will be considered good. Both, recall and precision values show that how the machine learning model works. The third performance parameter is the F1-score. It has a single evaluation metric that shows which algorithms performances are good. Due to F1-scores, we can quickly take decision which algorithm is best.

F1 Score = 2 PR/P + R

The overall results for current study are listed in Table 2. The fourth measure is sensitivity, which is the ratio of fail students correctly identified by the model [17]. Sensitivity quantifies how well the algorithm classified positive instances [11].

Sen = True Positive/(True Positive + False Negative)

The last evaluation method is specificity, which determines how well the algorithm classifies negative instances [11].

Sp = True Negative /(False Positive + Ture Negative)

Finally, we calculated the results for all the classifiers (Table 2). Sometimes bias affects model precision and accuracy.

Table 2. Overall Model Performances

LM	P	R	F1	Sen	Spe	Acc
GLM	100	61.9	76.5	61.9	100	90 %
DL	100	66.7	80.0	66.7	100	91 %
RF	100	71.4	83.3	71.4	100	92 %
GBT	100	61.9	76.5	61.9	100	90 %

Note: Learning Model (LM); Precision (P); Recall (R); F1-Score (F1); Sensitivity (Sen); Specificity (Spe); Accuracy (Acc); Generalized linear model (GLM); Deep Learning (DL); Random Forest (RF); Gradient Boosted Trees (GBT); All the above value is in percentage value performance .

The precision and accuracy value can mislead the researcher that the model achieved good performance in predicting fail students, when the data have unbalanced problem; therefore, current study checked other performance parameters such as F1-scores and recall (sensitivity) for the classifier performances.

As the current study wants to predict the fail students during the course assessment, therefore recall and F1-score are important performance parameter.

With the help of recall the current study can determine the fraction of students correctly identified as fail. Teachers can then give feedback to fail student, so that they could work hard for final exam and provide additional lecture material.

The current result shows that, DL and RF have high performance and appropriate classifiers for our data. In the experiment, the DL achieved a recall of 66.7 % and an F1-value of 80%, while the RF achieved a recall of 71.4 % and an average F1-value of 83.3%. The recall, F1 measure, accuracy and precision of all current study models are visualized in Figure 4, Figure 2, Figure 1 and Figure 3 respectively.

In a second experiment, we calculate the ROC curves for current study classifiers. The ROC curve displays the relationships between sensitivity (recall) and specificity. The ROC curve detects all of fail student's record in the dataset whose actually have fail student. The area under the curve (AUC) value will be 1. If the AUC of a classifier is high, it concludes that the classifier has high performance and the AUC value will be 0.5. If classifier has low AUC value, then the performance of classifier predicting fail student is poor. Figure 5 demonstrates that the performance of DL and RF classifiers is good because they have better ROC values than the other algorithms. Furthermore, the figure shows that the ROC curves of the other models are below than the other algorithms, it shows that the performance of those models is not good in our study.

The current study results conclude that both the DL and the RF are suitable algorithms for predicting fail students during course assessment. Instructors can employ these models to warn fail students in advance—before the students take the final exam for that course.

Fig 5. ROC curves of our Models

CONCLUSIONS AND FUTURE WORK

Predicting fail students is significant in web-based learning systems because it enables teachers to give intervention to students before a final exam. We used the data of VLE e-learning system. Before applying machine learning classifier, we formatted these data in a way acceptable for machine learning.

We then performed two experiments on VLE data. We trained all the current study classifiers to our data and check their performances on test data. The results show that the DL and the RF were the best algorithms for predicting fail students during VLE course, obtaining average F1-scores of 80% and 83.3%, respectively.

In future work, we will use the activities that students performed on each assessments of a social science course and then use clustering to recommend materials and activities for the fail students before they take the final exam. This result will help teachers to recommend good materials to students.

REFERENCES

1. Acharya, A. and Sinha, D., *Early Prediction of Students Performance using Machine Learning Techniques*. International Journal of Computer Application, 2014. 107(1): p.37–43.
2. Acikkar, M. and Akay, M.F., *Support vector machines for predicting the admission decision of a candidate to the School of Physical Education and Sports at Cukurova University*. Expert System with Application, 2009.36: p. 7228–7233.
3. Arora, Y., Singhal, A. and Bansal, A. *PREDICTION & WARNING: a method to improve student's performance*. ACM SIGSOFT Software Engineering Notes, 2014. 39(1): p.1–5.
4. *Introduction to deep neural network*. https://deeplearning 4j.org/neuralnet-overview/Accessed 25.5.18
5. Cawley, G.C. and Talbot, N.L.C. *Gene selection in cancer classification using sparse logistic regression with Bayesian regularization*. Bioinformatics, 2006. 22: p.2348–2355.
6. Chang, C. and Lin, C. *LIBSVM : A Library for Support Vector Machines*. ACM Trans Intell Syst Technol, 2011. 2: p.1–39.
7. Feng, M., Heffernan, N.T. and Koedinger, K.R. *Addressing the testing challenge with an online system that tutors as it assesses*. Journal of User Modeling and User-Adapted, 2009. 19: p.243-266.
8. Huang, B. and Mujumdar, A.S. *Use of a neural network to predict industrial dryer performance*. Drying Technology, 1993. 11: p.525–541.
9. Hsu, C.W., Chang, C.C. and Line, C.J. *A Practical Guide to Support Vector Classification*. BJU Int, 2008; 101:p. 1396–400.
10. *Technical Notes of Generalized linear model*. http://www.statsoft.com/Textbook/Generalized–Linear-Models/Accessed 25.5,2018.
11. Kotsiantis, S., Pierrakeas, C. and Pinellas, P. *Predicting Students' Performance in Distance Learning Using Machine Learning Techniques*. Appl Artif Intell, 2004: 18: p.411–426.
12. Mitchell, T.M. *Machine Learning*. Annual Review of Computer Science, 1997.
13. *Deep learning (DEL) Notes* .https:/en.wikipedia.org/wiki/Deep_learning.htm/Accessed 25.05.18.
14. Schaffer, C. *A conservation law for generalization performance*, ICML 94 Proceeding of the Eleventh International Conference on Machine learning, 1994: p.153-178.
15. Sharma, A.S., Prince, S., Kapoor, S. and Kumar, K. *PPS — Placement prediction system using logistic regression*. 2014 IEEE Int Conf MOOC, Innov Technol Educ. 2014: p.337–341.
16. Shishehchi, S., Banihashem, S.Y, Zin, N.A.M. et al *Review of personalized recommendation techniques for learners in e-learning systems*. 2011 Int Conf Semant Technol INF Retrieval, STAIR, 2011: p.277–281.
17. *Technical notes of Naïve Bayes Classifier*.http://www.statsoft.com/textbook/naive-bayes-classifier/Accessed 14.03.016, 2016.

18. Vahat, M., Oneto, L. and Anguita, D. et al. *A learning analytics approach to correlate the academic achievements of students with interaction data from an educational simulator*, 2015 In European Conference on Technology Enhanced Learning, 2015.11: p.352-366.
19. Zheng, J., Chen, Z. and Zhou, C. *Applying NN-based data mining to learning performance assessment*. 2011 Int IEEE Joint International Conference Computer Science and Information Technology (JICSIT).2011: p.1-5.
20. Kuzilek, J., Hlosta, M., and Zdrahal, Z. *Open university learning analytics dataset*. Scientific Data, 2017.4: p.170-171.