Machine Learning Based Engagement Prediction for Online Courses

Wanning Wang

Qingdao University of Science and Technology, Department of Information Sciences and Technology, 266061 No. 99 Songling Road, Qingdao, Shandong Province, China

Abstract. Within the constraints of the epidemic, the demand for distance learning in education is growing rapidly, and technological advances are opening up new possibilities for online education. This study investigates the performance of three machine learning models (decision trees, SVMs, and random forests) in predicting online course participation. To ensure the accuracy and generalizability of the results, the paper evaluated the models using k-fold cross-validation. Performance metrics such as accuracy, precision, recall and F1 score were used for comparison. The results show that the Random Forest model outperforms the other models on all metrics while the SVM model performs the weakest among the three models. Therefore, this study conducted a feature importance analysis specifically for the decision tree and random forest models to gain insight into the predictive power of individual features. This helps educators and course designers to develop strategies to improve engagement and retention. In summary, this study emphasizes the effectiveness of random forests in predicting engagement in online courses and highlights the potential of machine learning in improving the quality of e-learning environments. The findings can help optimize ongoing online education discussions and can guide future research in the field of e-learning.

1 Introduction

The new coronavirus outbreak at the end of 2019 has left many people stranded at home and unable to socialize as much as they would like. In order to be able to keep up with teaching and learning tasks, most students are at home utilizing online education platforms and resources. To ensure the quality of education, tools that can analyze and understand student engagement at scale are needed. Machine learning algorithms are able to process the massive amounts of data generated by online learning platforms to gain insights into student behavior. By analyzing the way students engage with course material, algorithms can customize content, pace, and pedagogy to the individual needs of the student, thus improving learning outcomes. Dias et al. proposed an application of the DeepLMS model to support online learning specific to online learning during epidemics, which inspired this study. The results of the study show that machine learning techniques are still highly predictive in online learning environments after an epidemic [1]. In addition, Mehta et al. (2022) developed a

Corresponding author: mhinkle75096@student.napavalley.edu

three-dimensional DenseNet self-attentive neural network for automatic detection of student engagement, demonstrating the strong potential of deep learning applications in this area [2]. Hussain and Wenhao Zhu et al. employed various ML algorithms to identify students with low engagement in an Open University (OU) social science course to assess the impact of engagement on student performance and investigated the relationship between student engagement and course assessment scores [3]. It was ultimately concluded that J48, Decision Tree, JRIP and Gradient Boosting classifiers performed better in terms of accuracy, kappa value and recall compared to other models. Based on this they developed a dashboard which is easy to use by the faculty members of the Open University. Nicholas R. Stepanek explores the feasibility of applying machine learning to categorize student posts according to level of engagement based on the IPCA framework [4]. The framework categorizes students' public behavior into four engagement levels: interactive, constructive, active, and passive. Ultimately, the analysis concludes that the linguistic characteristics of the textual works in the course under study largely determine the level of engagement, demonstrating the feasibility of its proposed machine learning approach, which can be generalized to many courses, and leveraging previous research to build analytical models using data from this machine learning approach. In this paper, Decision Tree, Support Vector Machine and Random Forest models are used to train the dataset respectively and the kFold method is utilized to uniformly verify the accuracy of model training. The best model is derived after comparing the accuracy, recall, and f1 score of each model, and the best model is optimized to produce the highest prediction.

2 Method

2.1 Decision Tree

As one of the powerful traditional machine learning tools, decision tree algorithms were first tried out. Decision tree algorithms work by connecting nodes one by one into a tree-like model for decision analysis. In classification problems, a decision tree divides a dataset by recursively selecting the optimal features, ultimately generating a complete tree structure in which each leaf node represents a classification result. This algorithm has the advantage of being computationally efficient and easy to understand and interpret.

Firstly, the tree module in sklearn library is imported to facilitate the creation of decision tree classes and methods and the specific decision tree classifier class is imported from the tree module. Then create an instance model of the decision tree classifier and fit the model using the training dataset X_train and the corresponding labels y_train. To optimise the model parameters, I use the fit() method.

Then the paper cross validates the data, KFold is a commonly used cross validation strategy, it will divide the data into n_splits of folds (folds) and then sequentially perform the training process once for each fold as a test set and the rest as a training set. Here the setup is to divide the data into 5 parts and randomly disrupt to ensure that each division is random and random_state is set to 42 to ensure repeatability of the results. cross_val_score function calculates the average accuracy score over all folds. Finally, the model was used to predict the test dataset X_test and the predictions were stored in the variable y_pred.

2.2 Support Vector Machine

The SVM module, which is a support vector machine classifier, was imported from the sklearn library. Then an instance of the support vector machine model with a linear kernel was created and its random state was set to 42 to ensure the reproducibility of the

experiment.this study performed a fitting operation on the model, i.e., the training data X_train and the corresponding label y_train were passed to the model for training by calling the fit() function.

Subsequently, the paper defined the KFold object kf, specifying the number of 5 folds, disrupting the order of the data, and setting the random state to 42. Next, the paper computed the cross-validation scores kf_scores, which were obtained by scoring the model's performance on different folds. After that, the paper predicted the test data using model.predict(X_test) and obtained the prediction y_pred.

In the last, the paper calculated several key metrics: accuracy, precision, recall, and F1 score. These metrics measure the model's ability to identify positive and negative examples, respectively, where the F1 score is a reconciled average of precision and recall, often used to balance the importance of the two.

2.3 Random Forest

In this study, the Random Forest algorithm was used for the construction and evaluation of classification models. Random forest is an integrated learning method that improves the accuracy and stability of prediction by combining multiple decision tree classifiers.

Firstly, this study imported the Random Forest classifier class and created a Random Forest model instance. The number of decision trees in the model was set to 100, and a random seed of 42 was also set to ensure repeatability of the results across runs. Next, the model was trained using the training dataset X_train and its corresponding label y_train. In order to assess the generalisation ability of the model, a 5-fold cross-validation was implemented in this study. In each iteration, the data were randomly disrupted and divided into five parts, four of which were used for training and the remaining one for validation. Accuracy scores were calculated as performance metrics for cross-validation.

After model training and validation, prediction was performed using the test dataset X_test and the prediction label y_pred was obtained. Subsequently, performance metrics of the model including accuracy, precision, recall and F1 score were calculated in this study. Finally, this study provides a visual analysis of the importance of the features of the model. By extracting the weight values of each feature in the model and ranking them according to the weight values, this study demonstrates the importance of each feature using bar charts. This step helps to understand which features play a key role in the prediction task.

3 Results

3.1 Dataset

The initial data for this study came from the online course engagement prediction dataset on the Kaggle website. This dataset collects user engagement metrics from online course platforms. There are nine variables, namely UserID, Course Category, Time Spent On Course, Number Of Videos Watched, Number Of Quizzes Taken, Quiz Scores, Completion Rate, Device Type, Course Completion (Target Variable). Since the data type of each variable is not the same, to simplify the algorithm's classification, the paper converted the string types to numeric types. To further clean the dataset for correct experimental results, the paper ignored the variable UserID and normalised the data using StandardScaler.

3.2 Graphical + textual explanation of the results of the experiment

The results obtained from the decision tree are shown in the Table 1, where the values of ACCURACY, PRECISION, RECOLLECTION and F1 are all around 0.9. Despite the good performance of the model, the recall is relatively low, indicating that there is still room for improvement. In comparison, the SVM results were not as good, with four values in the range interval of 0.7-0.8, achieving moderate to good performance on this dataset. Parameter tuning, feature engineering, or other optimization methods can be considered to further improve the model's performance. At the same time, it can be concluded that the random forest model has the best performance.

Table 1. Comparison of 3 models

Decision Tree	0.91666	0.9087	0.8839	0.8961
SVM	0.7916	0.7652	0.7035	0.7331
Random Forest	0.95611	0.9684	0.9221	0.9447

Simplifying the analysis on a better performing model improves the reliability of the model and thus makes the results more informative. In this way, this paper excludes the worst performing model, the SVM, to avoid the negative impact of models that may have large errors in the analysis of feature importance. The feature importance analysis of the other two models is shown in Fig. 1 and Fig. 2.

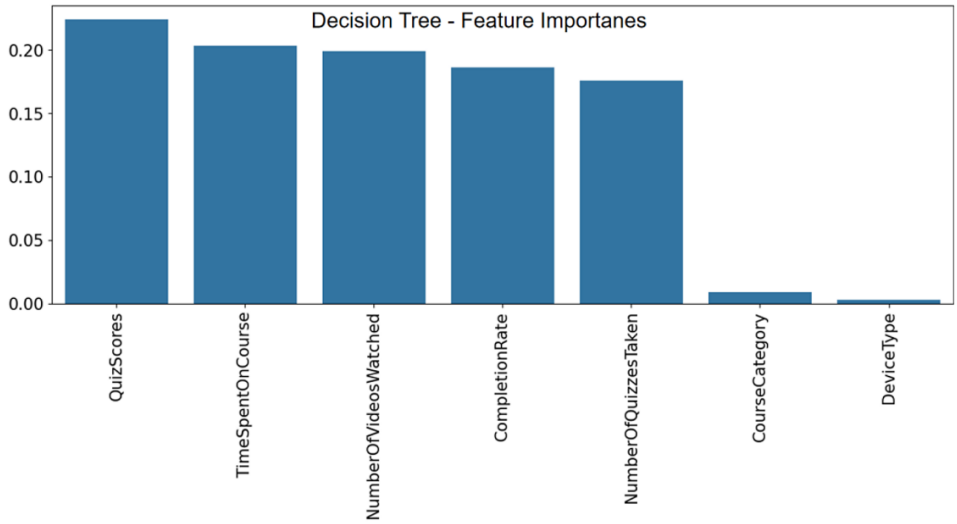

Fig. 1. Feature Importances of Decision Tree

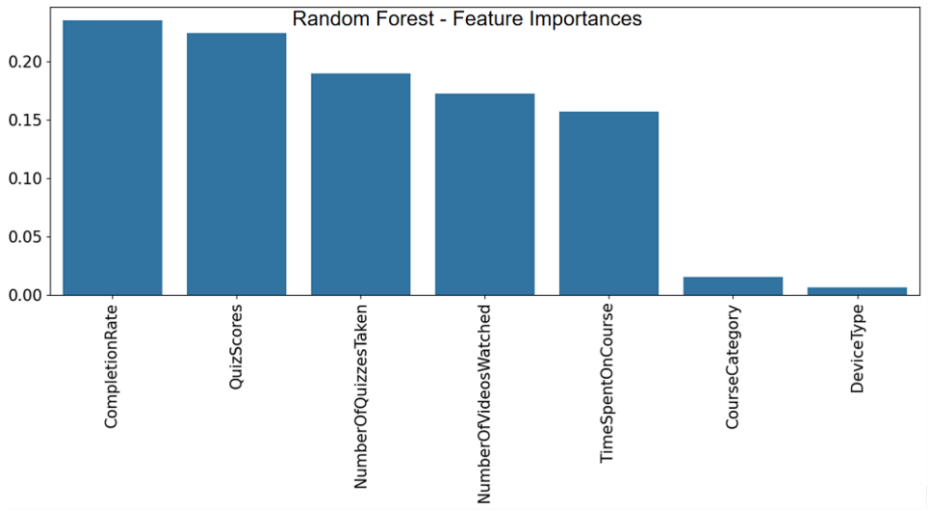

Fig. 2. Feature Importances of Random Forest

4 Discussion

There are some similarities between the findings and the machine learning-based online learning behavioral input assessment model proposed by Zhang Yue et al. [5]. This study draws on their ideas and improves the generalization ability of the model by extending the diversity of training data. This shows that the application of machine learning techniques in student engagement prediction is very effective and has a broad application prospect.

Second, the study found consistency with Lee et al.'s study exploring sustainable student engagement in e-Learning [6]. They emphasized the impact of factors such as student autonomy, the importance of feedback, and the suitability of learning resources on engagement. The study also confirms the importance of these factors and further demonstrates that these influences can be effectively captured and quantified through machine learning techniques.

Besides, the model produced by Selim et al.'s Hybrid EfficientNetB7, a combination of TCN, LSTM, and Bi-LSTM, is able to capture complex patterns and time dependencies in student interaction data, which is crucial for understanding engagement [7]. Research by Rajagopal et al. [8] and Alruwais and Zakariah [9] reinforces the idea that traditional machine learning methods such as support vector machines and decision trees remain important tools for predicting student engagement. These methods provide a solid foundation for analyzing engagement and can be further enhanced by feature engineering and optimization techniques. In addition, the study by Hew et al. emphasizes the importance of incorporating student reflections and qualitative data in machine learning-assisted analyses [10]. The practical application of this study is that it provides a powerful tool for online education platforms to predict student engagement. This is important for improving course design, development of personalized learning paths, and evaluation of teaching effectiveness.

5 Conclusion

The purpose of this study is to utilize machine learning techniques to predict student engagement in online courses and in the process explore the major factors that influence

student engagement. By analyzing data from multiple sources, the paper found several significant results.

However, there are some limitations of the study. First, the dataset is mainly derived from specific online learning platforms, which may lead to limited generalization ability of the model. Student behavioral patterns may vary across platforms, and future research could consider the fusion of multi-platform data to improve the generalizability of the model.

Future research should focus on extending the applicability of the model, exploring more potential influences, and considering the integration of multi-platform data. It is also possible to investigate how these predictions can be applied to educational practices, such as enhancing student engagement through real-time feedback and personalized recommendation systems. As online education continues to evolve, similar research will help to further enhance the quality and effectiveness of online learning.

References

1. S. B. Dias, S. J. Hadjileontiadou, J. Diniz, & L. J. Hadjileontiadis, DeepLMS: a deep learning predictive model for supporting online learning in the Covid-19 era. Scientific reports, 10(1), 19888 (2020).
2. N. K. Mehta, S. S. Prasad, S. Saurav, et al. Three-dimensional DenseNet self-attention neural network for automatic detection of student engagement. Appl Intell 52, 13803 – 13823 (2022).
3. M. Hussain, W. Zhu, W. Zhang, & S. M. R. Abidi, Student Engagement Predictions in an e-Learning System and Their Impact on Student Course Assessment Scores. Computational intelligence and neuroscience, (1), 6347186 (2018).
4. N. R. Stepanek, Towards Student Engagement Analytics: Applying Machine Learning to Student Posts in Online Lecture Videos. (2017).
5. Y. Zhang, S. M. Huang, Q. Wang, Y. H. Tang, Y. L. Yin. A machine learning-based evaluation model for online learning behavioral input. Computer Science and Applications, 13(8): 1596-1602 (2023).
6. J. Lee, H. Song, and A. Hong, Exploring Factors, and Indicators for Measuring Students' Sustainable Engagement in e-Learning. Sustainability, 11, Article No. 985 (2019).
7. T. Selim, I. Elkabani and M. A. Abdou, Students Engagement Level Detection in Online e-Learning Using Hybrid EfficientNetB7 Together With TCN, LSTM, and Bi-LSTM, in IEEE Access, vol. 10, pp. 99573-99583 (2022).
8. M. Rajagopal, B. Ali, S. S. Priya, W. A. Banu, M. G. M. Punamkumar, Machine Learning Methods for Online Education Case, 2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM), Chennai, India, pp. 1-7 (2023).
9. K. F. Hew, C. Qiao, & Y. Tang, Understanding Student Engagement in Large-Scale Open Online Courses: A Machine Learning Facilitated Analysis of Student's Reflections in 18 Highly Rated MOOCs. International Review of Research in Open and Distributed Learning, 19(3), 69-93 (2018).
10. N. Alruwais, & M. Zakariah, Student-Engagement Detection in Classroom Using Machine Learning Algorithm. Electronics (2079-9292), 12(3) (2023).