CANDIDATE NUMBER: 20

ACIT4530 DATA MINING AT SCALE: ALGORITHMS AND SYSTEMS PROJECT REPORT

ABSTRACT

Online learning or distance virtual learning has attracted several people mainly due to the fact that it allows to take up education irrespective of personal background and location. The main aim of imparting education is to help improve the learning outcomes of the students. However, the course completion rates for the online course programs are quite low. In this project, the main objective is prediction of student performance outcome. Predictive model would be developed through several experiments by applying various machine learning and data mining algorithms namely KNN, Random forest, Gradient Boost, Multi Layer Perceptron neural network model and RNN LSTM model. The dataset that would be used is Open University Learning Analytics Dataset (OULAD) which consists of demographic and click stream data for the students.

INTRODUCTION

In the recent years, there has been tremendous increase in the online learning or distance learning education globally. The online learning education platform provides the options of recorded video lectures, online assessments, discussion forums for clarifying doubts, live training session with the teacher through the internet. Due to the huge flexibility and level of comfortness that the online training offers, it has attracted several students to enrol in the online education. The online learning offers convenient opportunities and enormous resources for learning to pursue education for varied type of people around the world. This also promotes education for all age groups, level of education and professional background. Additionally, these days several educational institutions and universities enable students to take up for online courses. This is due to the fact that there could be huge number of students enrolled for online training when compared to the traditional education in-person. Also, the cost of online form of teaching is much cheaper when compared to the traditional way of education. All these factors greatly contributed in online learning booming significantly [4].

In traditional way of teaching, the teachers can interact with the students personally and can pay attention closely to the students. The teachers in this way can continuously assess the performance of students on multiple aspects including overall development and learning gain. The teachers can therefore take timely actions for improving the student's learning experience in achieving maximum learning gain. Thus, it leads to very few students failing the course or dropping from the education [4].

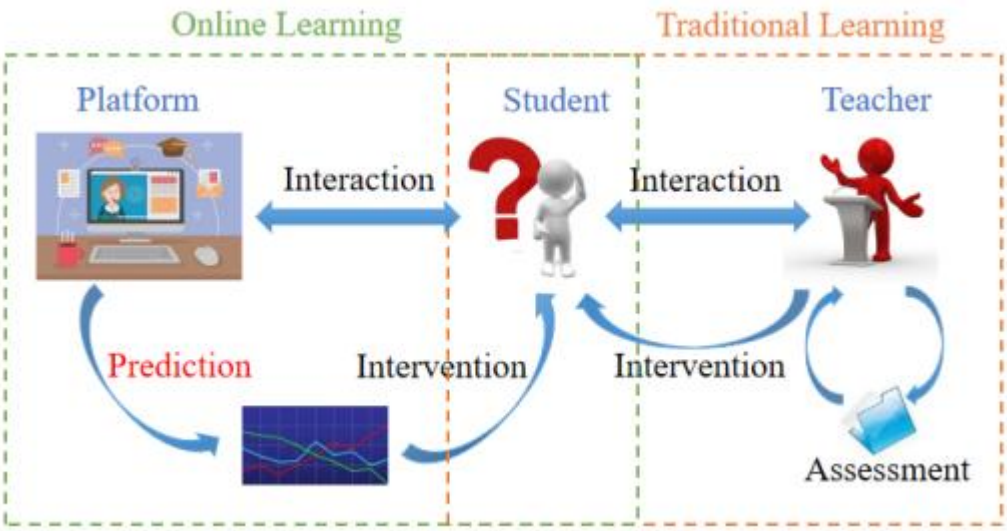


Figure 1 Comparison of Learning Process between online and traditional education systems [4]

However, the situation is quite different in case of online form of learning. There are less interactions between the students and the teachers. There are more number of students assigned to the teachers resulting in high student-teacher ratio. Also, the student diversity leads to the teachers not evaluating the learning gain of each student comprehensively in the online training platform. Due to all these reasons, there is a significant higher rates of students dropping out or failing the courses in online education than in traditional on premises education. For example, in Massive Open Online Courses (MOOCs) which is an extension of online learning technologies, the completion rates currently are low (0.7%-52.1% having median value as 12.6%), reported by Jordan 2015[14]. Similar is the case with the other online courses offered from the universities such as Open University UK and China (Jha, Ghergulescu and Moldovan 2019) [13][14].

Online education is independent form of education where the progress and the completion of the training completely depends on the motivation from the students. One of the challenges is that the students only get to interact with the computer system or application rather than the instructor or the teacher. Also, the online training has only access to the student demographic data and to the information about the student interactions with the training platform majorly in the form of click behaviours. Moreover, according to the Santos et. al 2014, the students drop out earlier in the courses , typically 75 percent of the dropouts occur in the initial weeks[12].

Therefore, there is requirement to find a way for assessing the performance of the students. This would ultimately give a platform and opportunity to give more attention to the students who are lacking the performance or are behind the progress track of the course. This would eventually reduce the student dropout and failure rates. Thus, there is a great need to determine the student's performance and assess their progress in the online training.

In the recent years, several research studies have been conducted on the extensive data being generated from various institutions offering online learning education. This available

educational data can be used to extract different patterns which could be useful meeting the sustainable development goals. In this regard, different machine learning techniques can be used and applied to extract the hidden information. To address this, a predictive model would be built and implemented to automatically predict the student performance and outcome and determine things such as success rate of the students. The OULAD dataset is used in this project by predicting the student performance in the examinations. The research question to be addressed is as follows: Can the machine learning and data mining techniques provide an insight into the student performance and thereby aid in taking the required necessary decisions for continuous improvement in the education training system?

RELATED WORK

The authors in the research paper [1] investigate ensemble methods, deep learning and regression techniques for prediction of student dropout and for the students who do not drop out from the course, predicting the final results pass or fail based on different group of attributes such as demographic info, assessment scores and VLE interaction information on the OULAD dataset. There were four different experiments conducted for both the drop out prediction and to classify the results. The machine learning models were built on different categories of the predictors including demographic info, assessment scores, VLE interactions and all attributes. However, the features such as the student id, code_module, module_presentation and exam_score were not included as part of the predictors.

For the results in the Drop Out predictions achieved in [1], the models created with the predictors as demographic information had AUC between 0.61 and 0.64. The models created with assessment scores as predictors achieved over 0.82 AUC and 0.84 for GBM and models based on VLE interaction features had AUC around 0.88 for GLM and 0.90 for DL, GBM and DRF on the validation data. Also, results show that models on all attributes achieved 0.01 higher AUC than the models based on VLE interactions only.

The models created for result classification based on demographics information achieved between 0.62 to 0.65 AUC on validation data. For assessment scores predictor, the AUC performance was 0.79 for DRF and 0.82 for GBM. The models achieved around 0.90 AUC based on VLE interaction features whereas models based on all attributes had 0.01 higher AUC than models based on VLE interactions alone.

The researchers in the paper [2] explored the time-series sequential classification problem of predicting the student's performance using a deep long short-term memory (LSTM) model for the OULAD (Open University Learning Analytics) dataset. The LSTM model for pass/fail classification job gave precision of 93.46% and recall of 75.79%. This model outperformed the baseline logistic regression and artificial neural networks by 18.48% and 12.31% respectively having 95.23% learning accuracy. The model within the first 10 weeks of student interaction through virtual learning environment (VLE) predicted pass/fail class with around 90% accuracy. The accuracy increased with additional weeks and the loss values tend to decrease with additional week-wise information. The academic performance for each student (pass/fail) was predicted with confidence of 69.69% obtained in the 1st week , 80.82% was achieved in 5th week and 95.23% was achieved in the last week. The accuracy of predicting whether student would pass or fail was over 85% from the 10th week. The results

showed that deep LSTM model had significantly improved performance when compared to the baseline models in predicting students at-risk.

The authors in the paper [3] present different machine learning algorithms for predicting student academic performance. The dimensional reduction algorithm using two algorithms Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) was applied on the OULAD dataset to reduce the dimension and extract the important features. The three supervised learning algorithms were used K-Nearest-Neighbours (KNN), Decision Tree, and Logistic Regression for predicting the target value which was predicting the result of the final examination. The work shows that the dimensional reduction algorithm followed by the prediction algorithm had reasonable accuracy for prediction of student performance. The classification accuracies were 75%, 99%, and 99.1% for Logistic Regression, KNN, and Decision Trees, respectively with PCA features and was 84.9%, 98.18%, and 99% with LDA features.

Citation	**Algorithms**		**AUC**	**Accuracy/ Precision/ Recall**	**Train Split**	**Test Split**	**Class Values**
[1]	Demographics	DL	0.634		75%	25%	Dropout/ No Dropout
		GLM	0.628				
		GBM	0.640				
		DRF	0.616				
[1]	Assessments	DL	0.826		75%	25%	Dropout/ No Dropout
		GLM	0.821				
		GBM	0.838				
		DRF	0.822				
[1]	VLE interaction	DL	0.900		75%	25%	Dropout/ No Dropout
		GLM	0.881				
		GBM	0.899				
		DRF	0.898				
[1]	All attributes	DL	0.907		75%	25%	Dropout/ No Dropout
		GLM	0.894				
		GBM	0.912				
		DRF	0.913				
[1]	Demographics	DL	0.627		75%	25%	Pass/Fail
		GLM	0.650				
		GBM	0.643				
		DRF	0.623				
[1]	Assessments	DL	0.806		75%	25%	Pass/Fail
		GLM	0.798				
		GBM	0.824				
		DRF	0.792				
[1]	VLE interaction	DL	0.897		75%	25%	Pass/Fail
		GLM	0.885				
		GBM	0.903				
		DRF	0.904				
[1]	All attributes	DL	0.918		75%	25%	Pass/Fail
		GLM	0.902				
		GBM	0.929				
		DRF	0.926				
[2]	LSTM		-	Accuracy-			Pass/fail

				95.23% Precision- 93.46% Recall- 75.79%			
[3]	Using PCA	Logistic Regression		75%	80%	20%	Distinction/ Pass/ Fail/ Withdrawn
		KNN		99%			
		Decision Trees		99.1%			
	Using LDA	Logistic Regression		84.9%			
		KNN		98.18%			
		Decision Trees		99%			
[5]	Learning Achievement Model	NNET1	0.95	0.950	70%	30%	Pass/Fail
		GBM	0.934	0.952			
		GLM	0.932	0.945			

Table 1 Summary of OULAD Research Studies

The authors in the research paper [4] analysed the online course performance for binary classification and four-class classification using the OULAD dataset. The data of year 2013 was used for training and the data of year 2014 was used for testing. The experiments demonstrated that the best model architecture for binary classification having 50 neurons for LSTM hidden size and a one layer FCN with 100 neurons. In case of four-class classification the best model had 100 hidden size for LSTM and one layer FCN with 50 hidden neurons. For both the classification, the more weekly data is introduced the models were better able to predict the student's outcome. Also, adding the demographic data boosted the performance of the model. Also, a poor performance of the proposed model DOPP is observed because the number of interactions with online platform at early stages is low. This is also because DOPP model based on deep model approach requires sufficient data for optimization. The DOPP model was used to study the intra-course and inter-course performance evaluation. For intra course, the model was trained on BBB, DDD and FFF for 2013B and 2013J periods and was tested on same course for 2014B using 20 weeks click data and demographics. The model obtained better results for intra-domain experiments with train and test course from same domain compared to inter-domain experiments with train and test courses from different domain.

In research paper[5], a two predictive model were developed namely at-risk student model and learning achievement model. The authors used two datasets namely Harvard dataset and OULAD dataset. The chi square test was utilized to filter the most significant features. The five machine learning models were applied namely NNET2, Random Forest, GLM, GBM and NNET1 to detect at-risk students for complete and reduced set of features. The F Measure results demonstrated highest performance for full and reduced set of features for the models GBM and NNET1 whereas RF and GLM had lowest performance. All the classifiers had good accuracy for both the models. The GBM yielded performance value of 0.894,

0.952 for first and second model respectively whereas RF model yielded value 0.866 in at-risk student framework achieved the lowest accuracy.

The study in research paper[6] predicts the performance of the student in particular course using personal information and sequential behaviour data with VLE. The authors have proposed a novel recurrent neural network (RNN)-gated recurrent unit (GRU) joint neural network where the missing stream data is filled. The authors performed a binary classification namely pass/fail for predicting the outcome of the student performance in the online course. The classes 'Pass' and 'Distinction' were considered as 'Pass' and 'Withdraw' class was ignored. The historical course data was used to predict the student performance in the current course. The architecture of the RNN-GRU model was implemented with two fully connected layers in demographics module with 128 neuron and three layers in prediction module from 384 to 1536 units. The RNN in both assessment and click module consisted of seven hidden layers with 256 units. The activation function used was Leaky Relu for each fully connected layer except the last layer in the prediction module and optimizer used was ADAM with learning rate 0.00002. The three kinds of time series deep neural network baseline models for processing the sequential learning data were used: RNN, GRU and LSTM. The experiments on the OULAD dataset showed that simple algorithms such as GRU and RNN had better results compared to complex LSTM model. The joint model proposed achieved over 80% accuracy in predicting for at-risk students at the end of semester.

The researchers in the study[9] used the demographic (static data) and student interactions with VLE Virtual Learning Environment for identifying the students at-risk. The most relevant VLE activity types are selected using Bayesian approach. The four predictive models were built using the demographic data for every week namely: Bayesian classifier, Classification and regression tree (CART), k nearest neighbours (k-NN) with demographic/static data and k-NN with VLE data. The precision had an increase from 50% at the beginning of the semester to 90% towards the end of the semester in prediction of at-risk students. The recall was constant at 50% throughout with 30% at the end because of incomplete results from the previous assessments.

SOLUTION

The dataset that would be used for prediction of the student performance would be OULAD dataset. A brief description about the dataset is as follows:

The OULAD dataset contains the student data from the courses presented at the Open University (OU) during 2013 and 2014. Open University offers distance learning at a large scale globally. Currently, there are nearly 170,000 students registered in different courses. The dataset contains the demographic data along with the aggregated click stream data of student interactions in the virtual learning environment (VLE). This dataset contains the details about 22 courses or module-presentations, 32593 students, along with their assessment results, daily summaries of student clicks (10,655,280 entries) and their interaction logs with the virtual learning environment [11].

The dataset contains a set of seven different CSV files as follows [11]:

student_info – It contains the student data along with their demographics and results. The student could have multiple rows if the student studied multiple modules. The file contains the below columns:

• *code_module* – identification code for the module for which student is registered.
• *code_presentation* – identification code of the presentation for which student is registered.
• *id_student* – student's unique identification number
• *gender* – gender of the student
• *region* – student's geographic region while studying the module-presentation.
• *highest_education* – student's highest student education level
• *imd_band* – It is UK specific social-economical indicator, indicates the band of the student residence during the module-presentation.
• *age_band* – student age band
• *num_of_prev_attempts* – number of attempts for module by student.
• *studied_credits* – total credits for modules studied by student currently
• *disability* – indicates whether the student has disability.
• *final_result* – final result of the student in the module-presentation.

student_registration – Registration data of each student for module presentations. File contains five columns:

• *code_module* – identification code for the module for which student is registered.
• *code_presentation* – identification code of the presentation for which student is registered.
• *id_student* – student's unique identification number
• *date_registration* – date of student registration for module presentation. The number of days is measured to the start date of module-presentation. If the value is negative e.g. -20, it means that student registered to module presentation 20 days before it started.
• *date_unregistration* – date of student unregistration from module presentation. It is number of days measured relative to the start of module-presentation. For students completed the course have the field as empty.

student_assessment – contains details about submissions and assessment results of each student. There is no result recorded in case the student does not submit the assessment. This file contains the following columns:

• *id_assessment* – identification number of assessment.
• *id_student* – student's unique identification number.
• *date_submitted* – submission date by student, it is number of days since start of module presentation.
• *is_banked* – status flag about assessment result that is transferred from previous presentation.
• *score* – score of the student in assessment. Score can range from 0 to 100. The score lower than 40 is denoted as fail.

student_vle – represents the interactional data between student and virtual learning environment. The file contains the following columns:

• *code_module* – identification code for module.
• *code_presentation* – identification code of module presentation.
• *id_student* – student's unique identification number
• *id_site* – identification number for VLE material.
• *date* – student's interaction date with the material, it is number of days since the start of module-presentation.
• *sum_click* – number of times the student interacts with the material on that day.

The Figure 2 shows the links, relationship and various attributes or columns for each of the file.

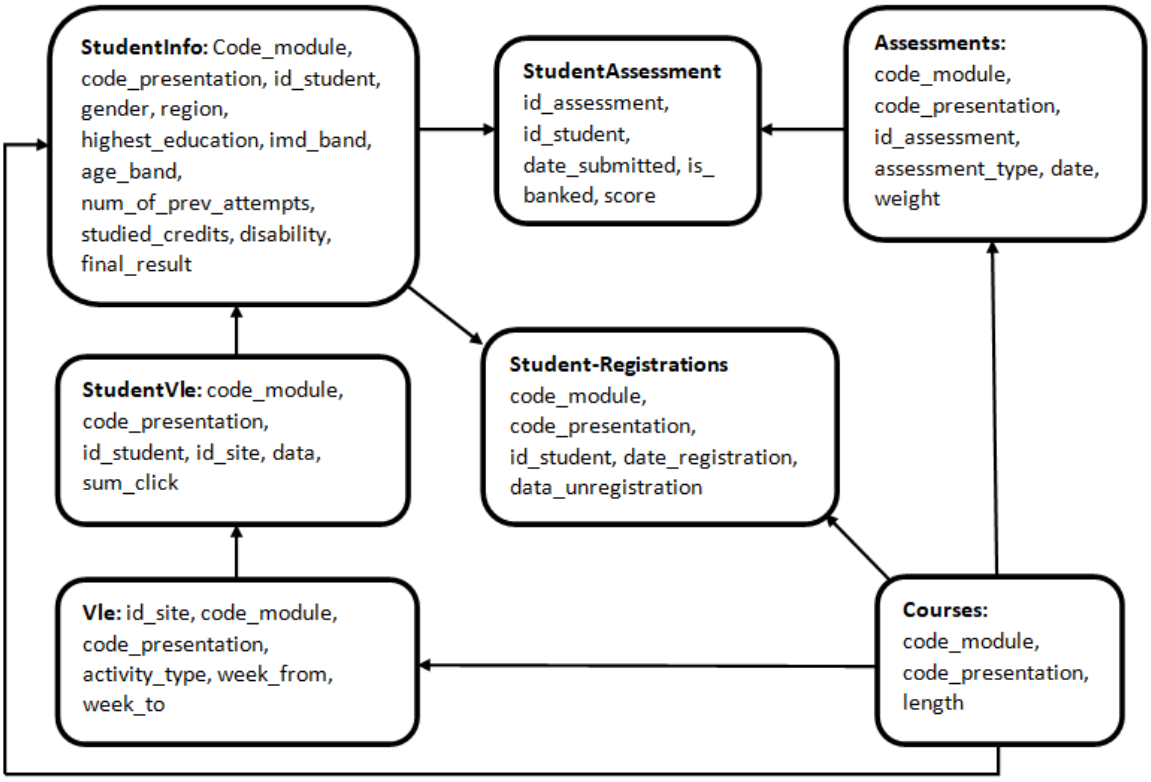


Figure 2 Detailed Structure of the Dataset[11]

vle – This file consists of the data in the form of html pages and pdf files about materials and available Virtual Learning Environment (VLE). The file contains the following columns:

• *id_site* – identification number of the material.
• *code_module* – identification code for module.
• *code_presentation* – identification code of module presentation.
• *activity_type* – defines the role associated with the module material
• *week_from* –week from which the material would be used.
• *week_to* – week until which the material would be used.

assessments – contains the details for all assessments planned in module-presentations.
• *code_module* – identification code of the module
• *code_presentation* – identification code of the presentation
• *id_assessment* – identification number of the assessment.
• *assessment_type* – defines the type of assessment namely Tutor Marked Assessment (TMA), Computer Marked Assessment (CMA) and Final Exam (Exam).
• *date* – contains the final submission date of the assessment. It is calculated as the number of days since the start of the module-presentation assuming starting date of the presentation has number 0 (zero).
• *weight* – weight of the assessment in %. Exams have the weight 100%; sum of all other assessments is 100%.

courses – Lists all available modules and their presentations in the dataset

• *code_module* – code name of the module
• *code_presentation* – Consists of the year and letter for presentations, B for presentations starting in February and J for presentations starting in October.
• *length* – length of the module-presentation in days.

EXPLORATORY DATA ANALYSIS AND VISUALIZATION

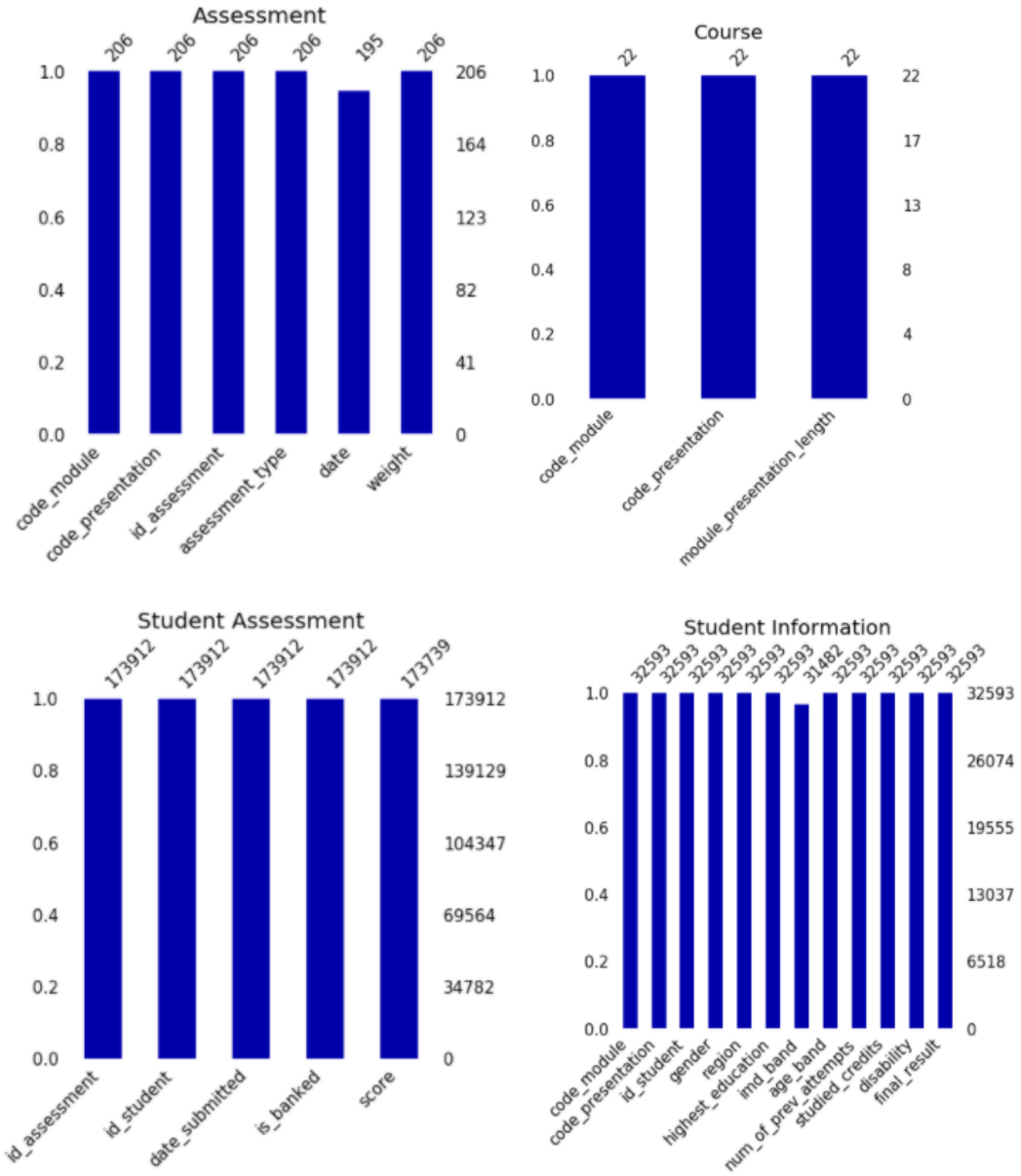
The Table 2 describes the features, number of rows and columns contained in each table for OULAD dataset.

Table name	**Rows, Columns**	**Feature Names**
Assessment	(206,7)	[code_module, code_presentation, id_assessment, assessment_type, date, weight, module_presentation]
Course	(22,4)	[code_module, code_presentation, module_presentation_length, module_presentation]
student_assessment	(173912,5)	[id_assessment, id_student, date_submitted, is_banked, score]
student_info	(32593,13)	[code_module, code_presentation, id_student, gender, region, highest_education, imd_band, age_band, num_of_prev_attempts, studied_credits, disability, final_result, module_presentation]
student_reg	(32593,6)	[code_module, code_presentation, id_student, date_registration, date_unregistration, module_presentation]
student_vle	(10655280, 7)	[code_module, code_presentation, id_student, id_site, date, sum_click, module_presentation]
Vle	(6364, 7)	[id_site, code_module, code_presentation, activity_type, week_from, week_to, module_presentation]

Table 2 OULAD Dataset Description

Checking Missing or Null values

It is important when analysing certain dataset to know if there are any values which are missing or have null values. The missing values are represented as NaN(Not a Number) value. The Figure 3 below shows the missing values for the student data.



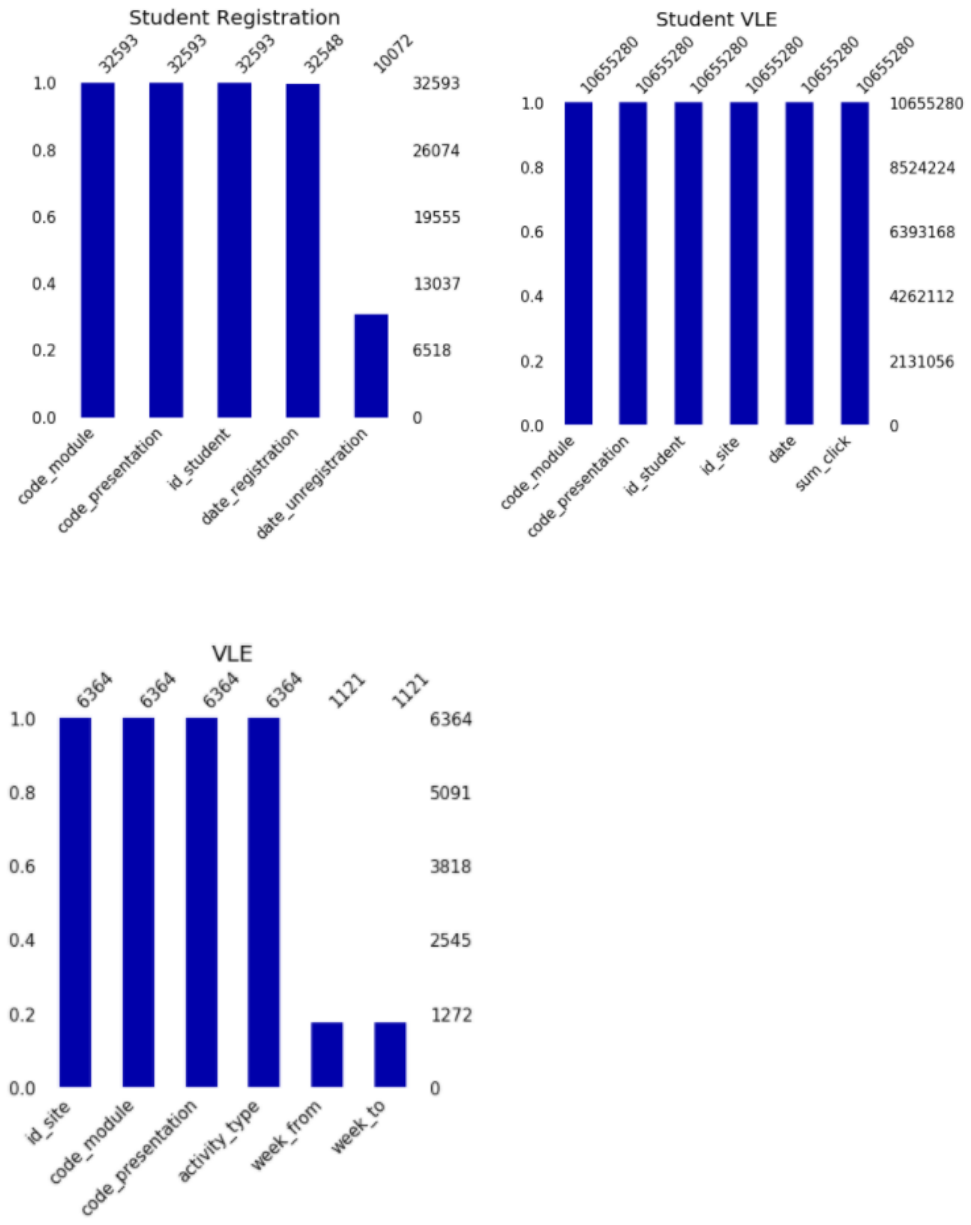


Figure 3 Missing Values for student data

From the Student Registration graph above in Figure 3, 70% of the rows are not containing data for date_unregistration. This indicates that 70% of the students don't withdraw from the modules. The consistency of the students that are unregistered is checked against the final_result. The final_result of the students that are unregistered should be ‘Withdrawn’. The student_info table is corrected with final_result value for this condition.

Also,in the VLE graph above, nearly 80% of the values are missing for the columns week_from and week_to. Therefore, these columns are not significant in the dataset analysis.

Cleaning and Modifying the Dataset The missing values in all the seven dataframes are dropped using dropna method.

After analysing the dataset, the module is always identified with the Code module and code presentation. Therefore, a new feature or column namely 'module_presentation' is created with the combination of the above two columns in the following dataframes:course, assessment, vle, student_info, student_reg, student_vle.

Dataset Analysis and Visualization

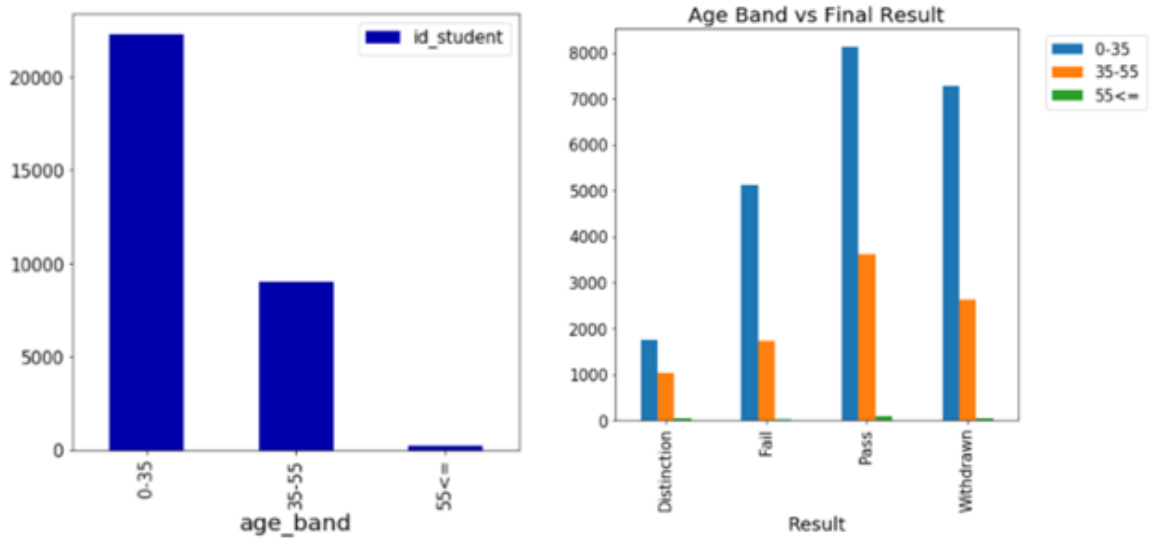


Figure 4 Age Band versus Final Result Graph

From the Figure 4, most of the students fall in the age group of 0-35 years. The figure shows the distribution of students into the final result based on the age groups.

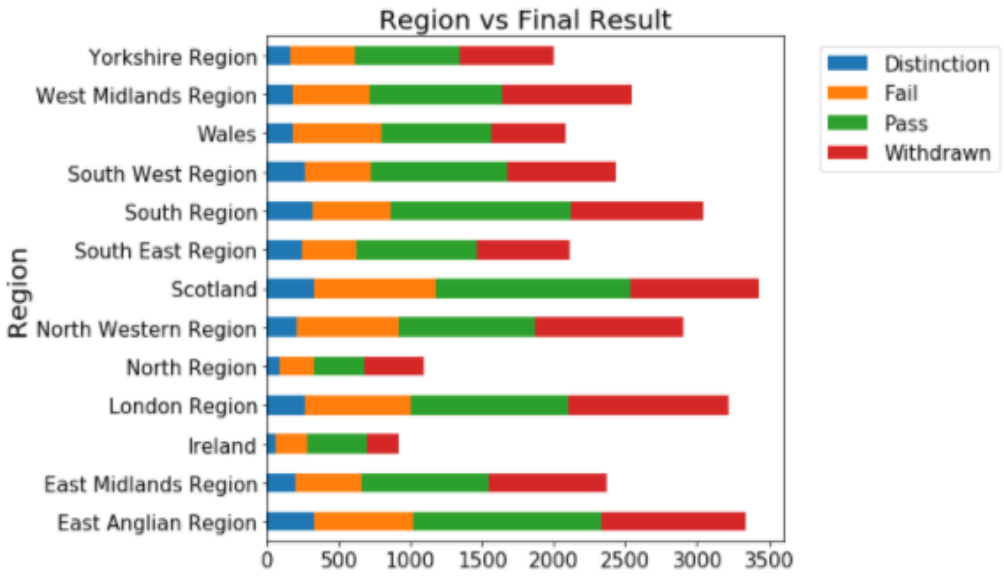


Figure 5 Region versus Final Result Graph

The Figure 5 shows the plot of students in different regions versus the final result obtained. From the figure, there are comparatively less students in the North Region and Ireland when compared to the other regions.

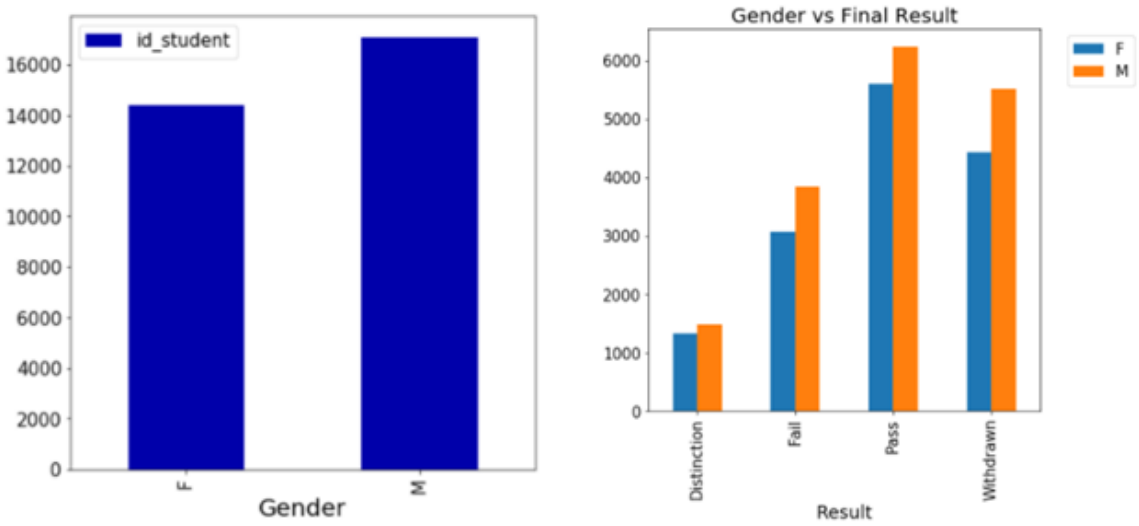


Figure 6 Gender versus Final Result Graph

The Figure 6 shows the distribution of students according to the gender and a graph of gender versus final result. The number of male students is little higher than the number of female students.

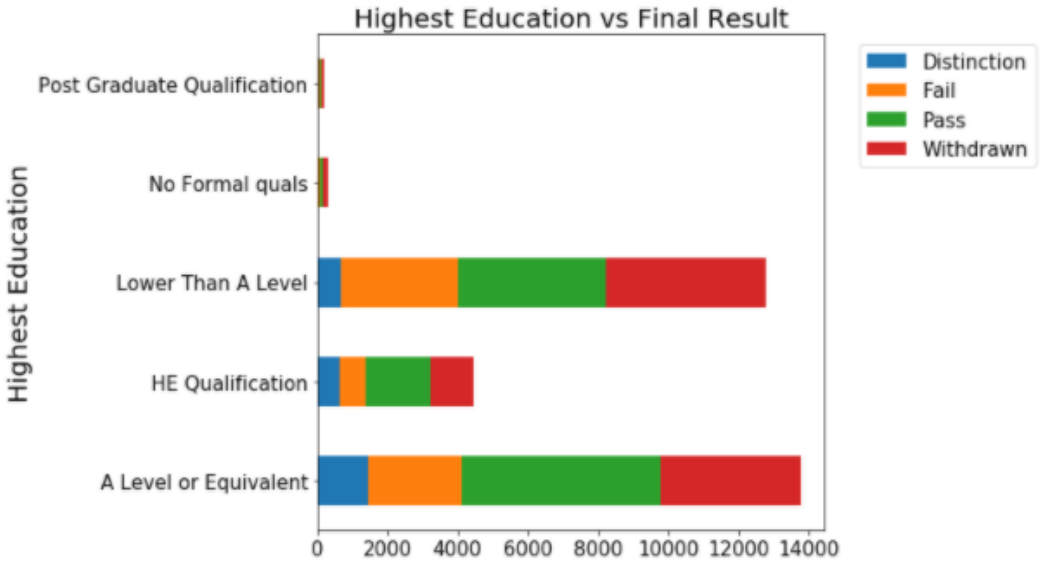


Figure 7 Highest Education versus Final Result Graph

The Figure 7 shows the number of students with higher education. The students with No Formal qualifications or Post Graduate Qualifications are very less.

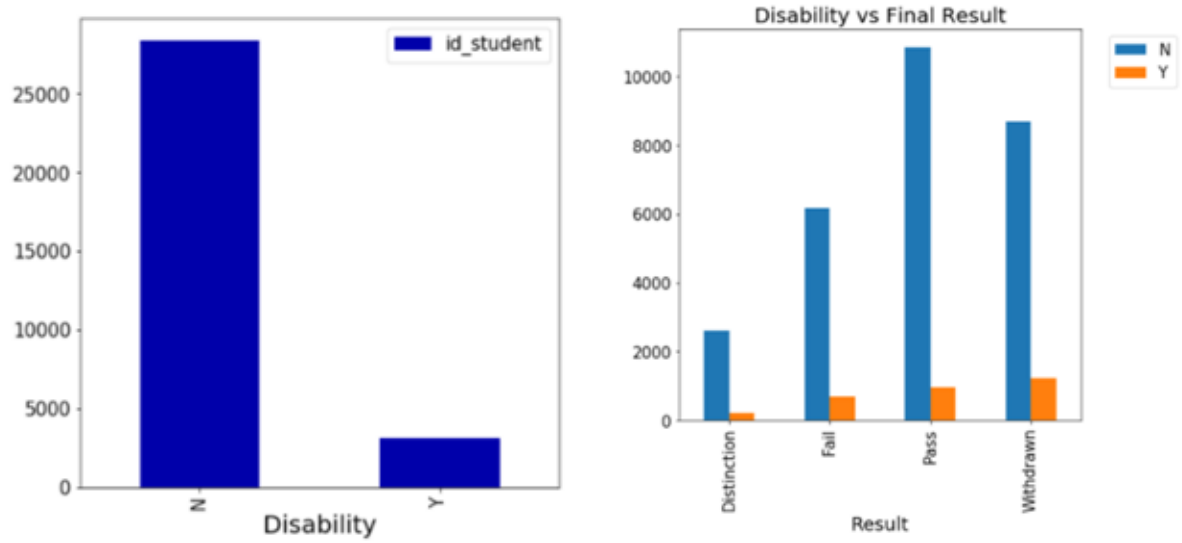


Figure 8 Disability versus Final Result Graph

The Figure 8 shows the number of students with disability and their respective final result. number of students with disability are relatively less and the students with disability have higher percentage of withdrawing from the course.

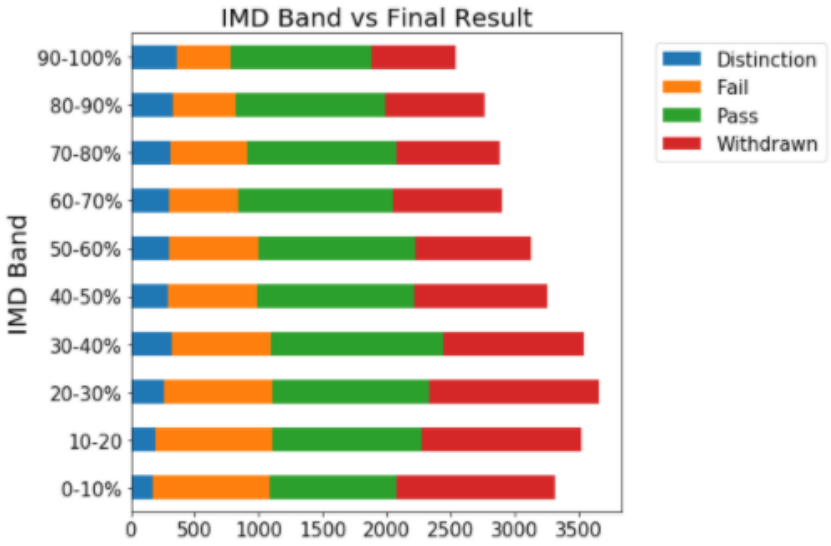


Figure 9 IMD Band versus Final Result

The Figure 9 shows the graph of students in different IMD band versus the final result. There distribution of students with all four results namely Distinction, Pass, Fail and Withdrawn in all the IMD band category.

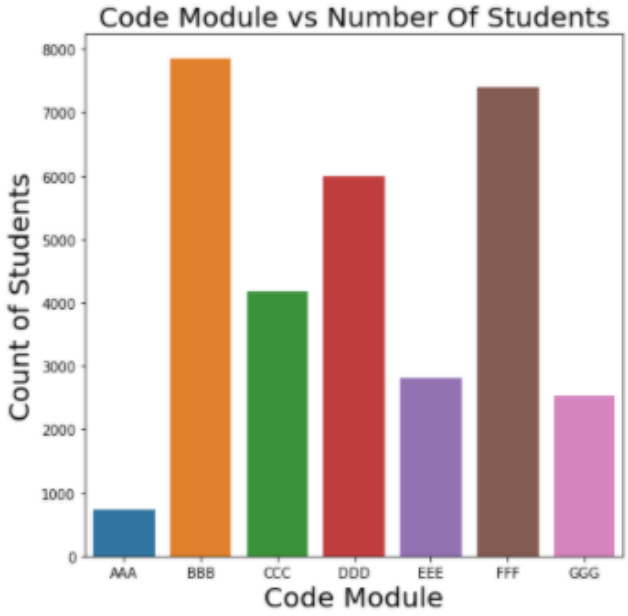


Figure 10 Number of students in Code Module

The Figure 10 shows the count of students in different Code Module namely: AAA, BBB, CCC, DDD, EEE, FFF and GGG. Maximum students registered for the courses BBB and FFF whereas the module AAA had very less students registered.

Dataset Mining Methodology

The below steps also illustrated in Figure 11 are followed for implementing the various classification model algorithms as illustrated in the figure.

- The input data is collected, visualized and analysed.
- The features which are most important and relevant in determining the target class are selected using different feature selection methods.
- The dataset dimension reduction is done by eliminating the least significant features
- Feature Scaling is performed using standardization.
- The dataset is splitted into two sets training sets consisting of 70% and testing set consisting of 30%.
- Different machine learning and data mining classification algorithms are applied on the dataset.
- Finally, the model performance is evaluated by various evaluation metrics parameters

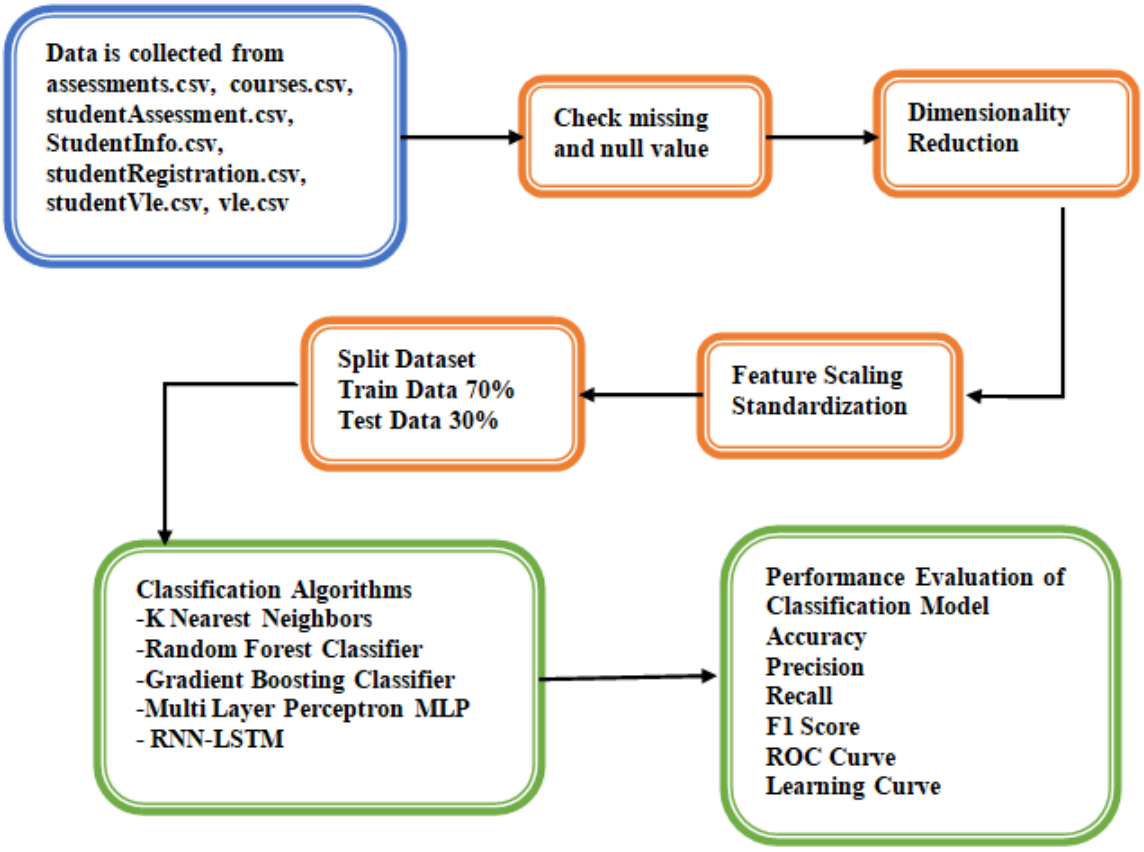


Figure 11 Machine Learning and Data Mining Algorithm Methodology

Pre-processing the dataset In this step, the important features are selected for feature set in order to generate predictive model

A new column ‘sum_of_sum_click’ is created in student_vle data by aggregating the sum of sum_click features by grouping id_student, module_presentation. Similarly, a new column

‘avg_score’ is created in student-assessment data by calculating the mean of the score values for the student.

The following feature set described in Table 3 is selected for applying machine learning algorithms.

Dataset	Selected features
student_info	module_presentation, id_student, highest_education, num_of_prev_attempts,studied_credits,disability,final_result
student_reg	id_student, module_presentation,date_registration,date_unregistration
student_assessment	id_assessment,id_student,date_submitted,avg_score
student_vle	id_student, module_presentation, sum_of_sum_click

Table 3 Selected Features from Dataset

The above four datasets are then merged using pandas merge function by merging the dataset tables on common key fields. After merging, the final dataset contains 32480 entries.

The Figure 12 shows the distribution of the dataset for the final result among the students

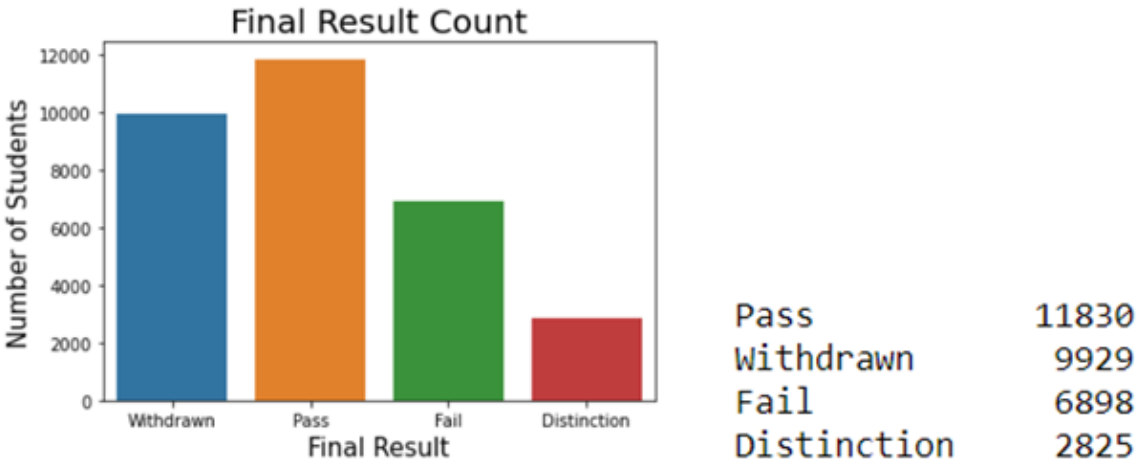


Figure 12 Final Result Count Distribution

Normalizing the Dataset

The dataset would be modified so as to make all the features as numerical. The values for the below features would be modified to numerical values as below in Table 4.

Feature/Column	Existing value	New value
disability	N	1
disability	Y	2
final_result	Fail	1
final_result	Withdrawn	2
final_result	Pass	3
final_result	Distinction	4

highest_education	No formal quals	1
highest_education	Lower Than A Level	2
highest_education	A Level or Equivalent	3
highest_education	HE Qualification	4
highest_education	Post Graduate Qualification	5
module_presentation	AAA_2013J	1
module_presentation	AAA_2014J	2
module_presentation	BBB_2013B	3
module_presentation	BBB_2013J	4
module_presentation	BBB_2014B	5
module_presentation	BBB_2014J	6
module_presentation	CCC_2014B	7
module_presentation	CCC_2014J	8
module_presentation	DDD_2013B	9
module_presentation	DDD_2013J	10
module_presentation	DDD_2014B	11
module_presentation	DDD_2014J	12
module_presentation	EEE_2013J	13
module_presentation	EEE_2014B	14
module_presentation	EEE_2014J	15
module_presentation	FFF_2013B	16
module_presentation	FFF_2013J	17
module_presentation	FFF_2014B	18
module_presentation	FFF_2014J	19
module_presentation	GGG_2013J	20
module_presentation	GGG_2014B	21
module_presentation	GGG_2014J	22

Table 4 Modification in the OULAD dataset

Feature Scaling

This is required to ensure that all the variable values are in the same range or in same scale so that there are no dominant features than others. The Figure 13 and Figure 14 shows the correlation of the numerical features with the final result. As seen, the features ‘avg_score’, ‘sum_of_sum_click’ and ‘highest_education’ have a very high correlation with the final result target value.

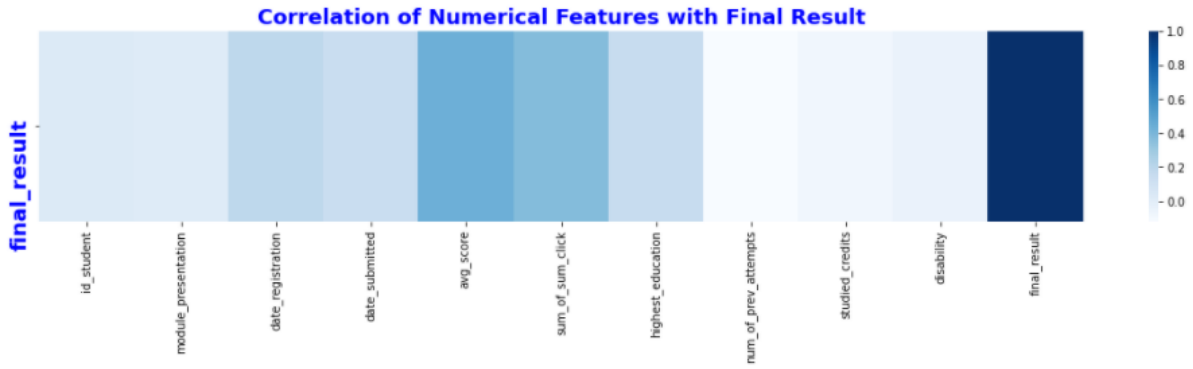


Figure 13 Correlation of Numerical Features with Final Result

	final_result
final_result	1.000000
avg_score	0.436766
sum_of_sum_click	0.373714
date_registration	0.191258
highest_education	0.156742
date_submitted	0.147096
id_student	0.034045
module_presentation	0.022444
disability	-0.041371
studied_credits	-0.074657
num_of_prev_attempts	-0.117697

Figure 14 Correlation of features with Final Result

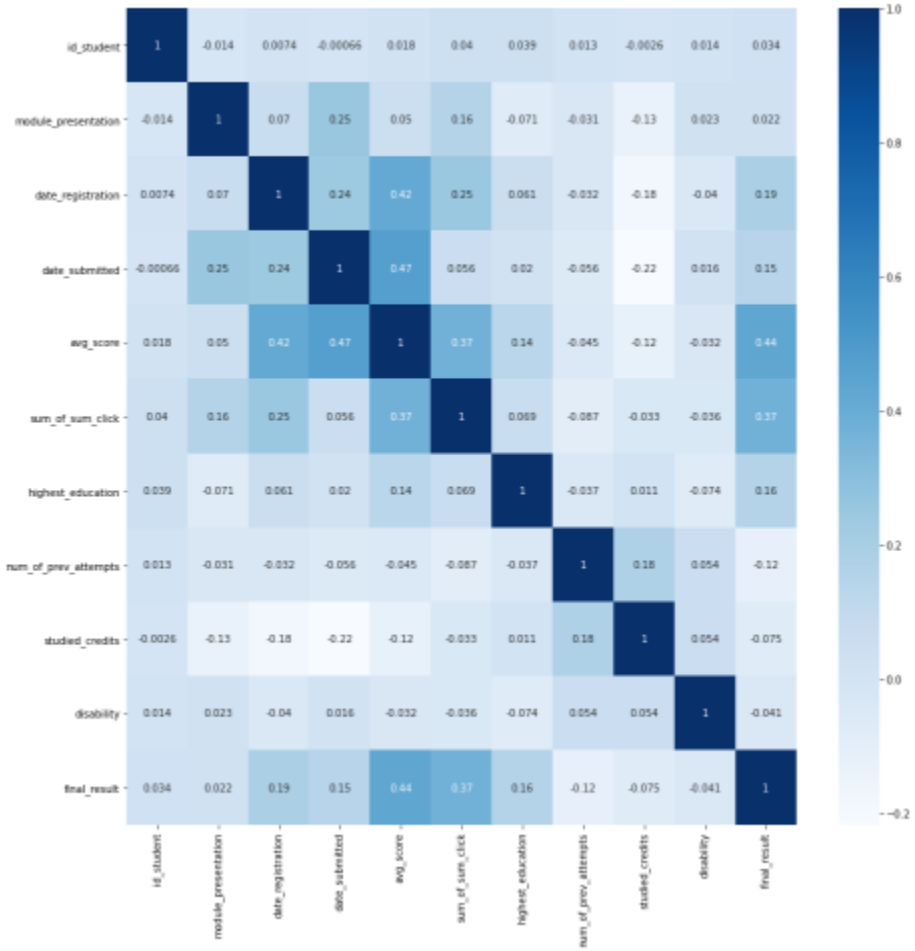
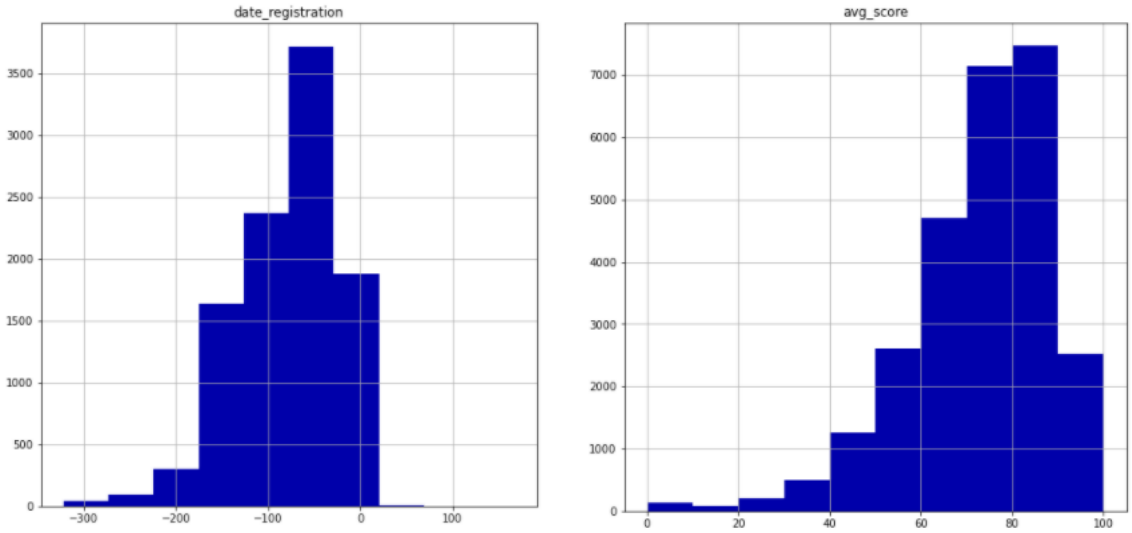


Figure 15 Heatmap of different features in OULAD dataset

The graph in the Figure16 represent the histogram plot for all the numerical features.



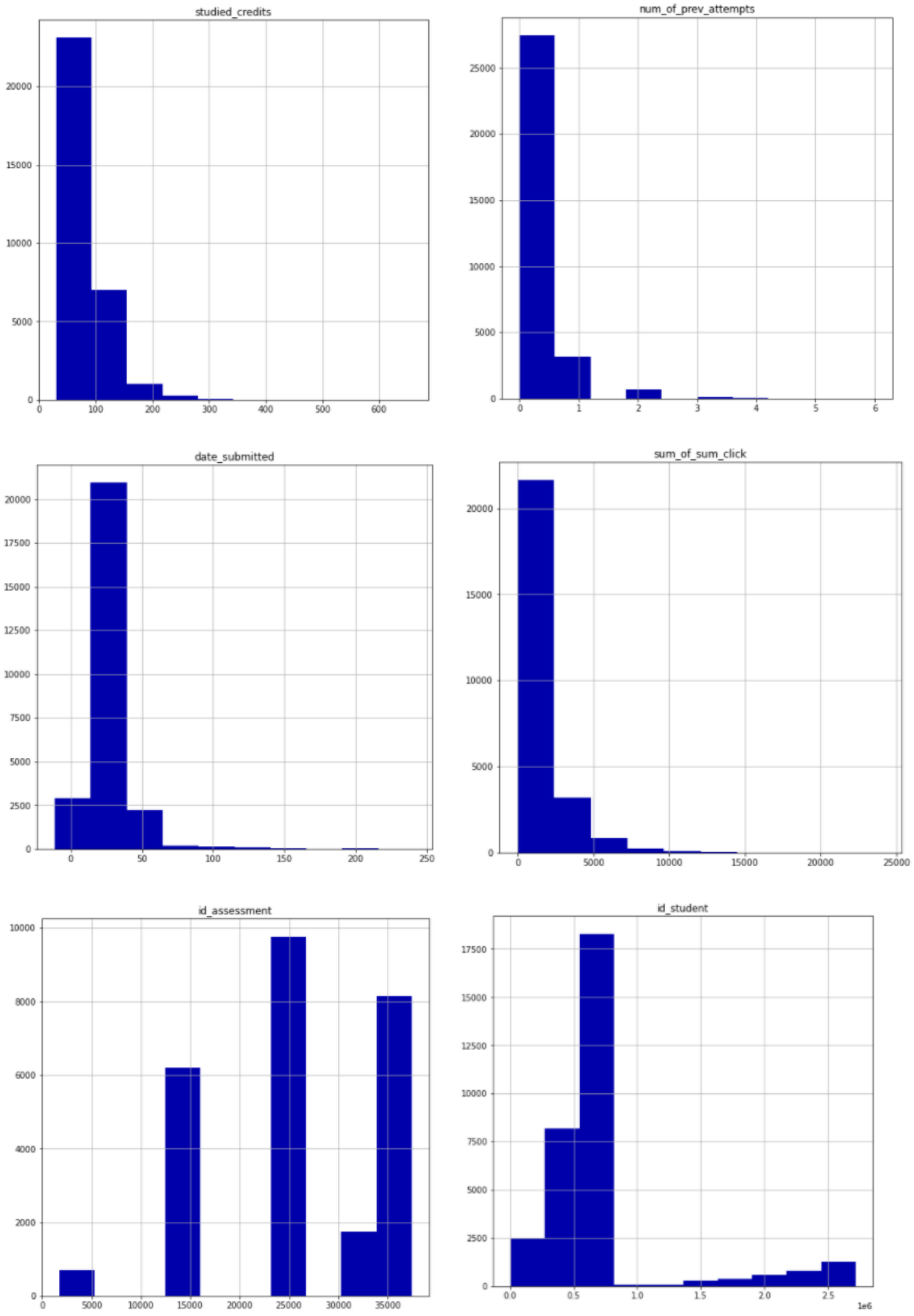


Figure 16 Histogram Plot of features in OULAD dataset

Train Test Split The dataset is splitted into train dataset and test dataset with 70% and 30% ratio from the given dataset. The train dataset would be used for training the different models and the unknown test dataset would be used for testing the model. This ensures better calculation of performance of the model.

Building the predictive models The goal of for this dataset analysis is to predict student performance with least prediction error. The prediction of the student performance are classified into classes as Distinction, Pass, Fail, Withdrawn is a multiclass classification problem where the output variable which is final result in this case can be classified into any one out of the three classes.

The below machine learning and data mining algorithms would be applied on the OULAD dataset to predict the student performance with the highest accuracy.

Machine Learning Models: The following machine learning models would be implemented with hyper parameter tuning method on the OULAD dataset.

K-Nearest Neighbours: It is a supervised learning algorithm used for classification problem that assumes similar items exist close to each other [15].

Random Forest: It is an ensemble learning classification method operating by constructing several decision trees and calculating classification [16].

Gradient Boosting: It is machine learning algorithm for solving classification problems, producing ensemble of weak prediction model that is decision trees building models in stage wise pattern[17].

Neural Network Models: Tensorflow and Keras libraries are used to implement the following neural networks for OULAD dataset. For implementing deep learning models using neural networks with tensorflow.keras API, there are five steps as following [18]

Defining the model- This involves selecting the model and choosing the network topology architecture which means define the layers of the model, configure each layer with number of nodes and activation function, and connecting the layers.

Compiling the model- In this step, a loss function is selected that needs to be optimized such as mean squared error or cross entropy. Also an algorithm for optimization needs to be selected such as Stochaistic Gradient descent or Adam.

Fitting the model - This includes selecting the training configuration such as number of epochs that is number of loops through dataset and batch size which is number of samples used in an epoch. The optimization algorithm selected is applied during training in order to minimize the loss function and updating model using back propagation of error algorithm.

Evaluating the model – This requires choosing holdout set which is the data not included in the training dataset to get the unbiased results in order to evaluate the model performance in making predictions on new data.

Making predictions – This step includes predicting the output values for the new data where the target values are not known.

Multilayer Perceptron: It is a standard fully connected neural network model. This network model consists of layers of nodes where every node is connected to each output from the previous layer and also the output of every node is also connected to all the inputs for the nodes present in the next layer[19].

Multilayer Perceptron model is created with either one or several Dense layers. This type of model is more suitable for tabular data present in a table or spreadsheet where there is one row and one column for each variable. MLP can be used for binary classification, multiclass classification and regression problems[19].

Below is the architecture of the multi layer perceptron model built using keras.

Model is initialized with Sequential: *model=Sequential()* and expects rows of data with 9 variables as input_dim=9 is the argument.

There are 3 Dense layers as follows:

1) First hidden layer contains 9 nodes and uses activation function as relu and initial kernel is set to random normal.

model.add(Dense(9, input_dim=9, activation='relu',kernel_initializer='random_normal')

2) Second hidden layer contains 5 nodes, activation function is relu and initial kernel is set to random normal.

model.add(Dense(5, activation='relu', kernel_initializer='random_normal'))

3) Output Layer has four nodes, activation function is softmax and initial kernel is set to random normal.

model.add(Dense(4, activation='softmax', kernel_initializer='random_normal'))

The learning optimizer used is Adam, number of training epochs are set to 100 and the model is trained in batches of 16 samples and the loss function is defined by categorical crossentropy.

RNN LSTM – Recurrent Neural Networks RNN are useful for operating on the sequences of data especially in the areas of natural language processing problems where the input consists of sequences of text data. It is also effective in time series forecasting and speech recognition. Long Short-Term Memory network or LSTM is a popular variant of RNN where it accepts sequence of data as input to make predictions for class labels or predicting next values in the sequence[20].

In this problem, the performance of the models could be higher when the course is nearing the completion due to the click and interaction data with VLE and also higher number of assessments taken. The model can therefore be trained and assessed at different time intervals with the clickstream data and score data.

Below is the architecture of the RNN LSTM model built using keras.

Model is initialized with Sequential: *model=Sequential()* and expects rows of data of sequence data with first set of variables consisting of 3 variables(code_ presentation, id_ student, date) and second containing 2 variables (sum_ of_clicks, score)

The model consists of the below layers as follows:

1) Input layer
2) First masking layer uses mask value as -1
3) Second hidden LSTM layer with 16 nodes with activation function as tanh for learning non linear functions
4) Drop out layer setting 40 percent of input units to 0 to prevent overfitting
5) Output layer having single node with activation function as sigmoid.

The learning optimizer used is Adam, number of training epochs are set to 100 and the batch size is defined as 32.

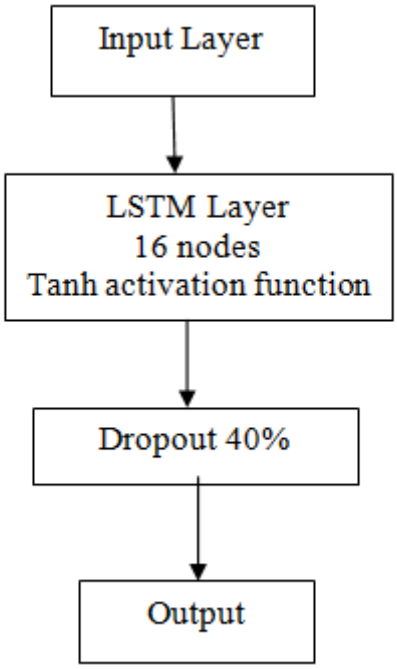


Figure 17 LSTM Network Architecture

For building LSTM model, the data was slightly tweaked to create time series data.

The final dataset consisted of code_module, code_presentation, id_student, date, score, sum_click , final_result which was generated by merging the respective columns from different data frames of the student data. The column ‘date’ had the value in the range of -23 to 267 for course modules which denoted the offset of the day based on the course registration date by the student.

However, there were records for the date column which were missing for students. Since the dataset is quite huge consisting of data for 32593 students. Therefore, in this case the LSTM

model was built on a smaller dataset considering the code module as AAA and code presentation as 2013J. This building of LSTM model could be however extended to the entire dataset by preparing the dataset in the required format.

Challenges in developing LSTM Model The missing data generation logic for date value was developed and implemented. However, the dataset couldn't be created properly and some of the values for the date field were missing and few values were additional. Therefore, due to time constraints couldn't finish the building of LSTM Model for this dataset problem.

Performance Evaluation metrics

After training the model the most important part is to evaluate the classifier to verify its applicability. Below methodologies and parameters are used to assess the model [21]

Holdout method - In this method, the available dataset is divided into two groups namely train and test data which comprises of 70% and 30% of data respectively. The train data is used for training the model and the remaining test data is used for determining the correctness of prediction [21].

Cross-validation - K-fold cross validation is a evaluation method to ensure that the model is not over-fitted. The dataset is partitioned into k mutually exclusive subsets randomly almost having same size and out of them one is picked up for testing and others for utilized for training. The entire process is repeated for k folds. This process is iterated throughout the whole k folds [21].

Precision - It is also known as positive prediction value which is fraction of relevant instances among retrieved instances[22].

Recall - It is also known as sensitivity is the fraction of relevant instances among those that were actually retrieved[22].

Confusion Matrix - It displays the classification model prediction results. It helps to understand how far is the model is correct and insights into type of errors[23].

ROC curve (Receiver Operating Characteristics) - ROC curve is a visual representation for comparing the classification model that displays the graph of between the true positive rate and false positive rate. The area under the ROC curve gives the accuracy of the model. The more the model is away from the diagonal, the more it is accurate. An area of 1.0 denotes that the model is having perfect accuracy [21].

Learning Curve -These plots help to show developmental changes in performance during learning. It is also used to detect based on the train and test/validation dataset as under-fitted, over-fitted or well-fitted model [24].

EXPERIMENTAL RESULTS

The table below shows the model accuracy for the machine learning algorithms K Nearest Neighbors, Random forest and Gradient Boosting algorithm for the base model and the

accuracy after applying hyperparameter tuning. The results show that the accuracy for the models increased slightly after applying hyperparameter tuning using randomized search.

Machine Learning Algorithm	Base Model Accuracy	Accuracy after Hyper Parameter Tuning
K Nearest Neighbour	0.727	0.734
Gradient Boosting	0.814	0.815
Random Forest	0.809	0.815

Table 5 ML model accuracy of base model and accuracy after Hyperparameter tuning.

ML Algorithm	Accuracy	RMSE	Precision	Recall	F1 score
K Nearest Neighbour	0.73	0.836	0.73	0.61	0.63
Gradient Boosting	0.815	0.738	0.79	0.74	0.76
Random Forest	0.815	0.738	0.79	0.74	0.76
Multi Layer Perceptron	0.90	-	-	-	-

Table 6 Performance evaluation metrics for different classification models

The table below shows the number of samples for each of the target classes in the yTest (Test dataset).

Final Result	Number of test samples
Fail	2147
Withdrawn	2892
Pass	3538
Distinction	868

Table 7 Number of samples in Test dataset for final result

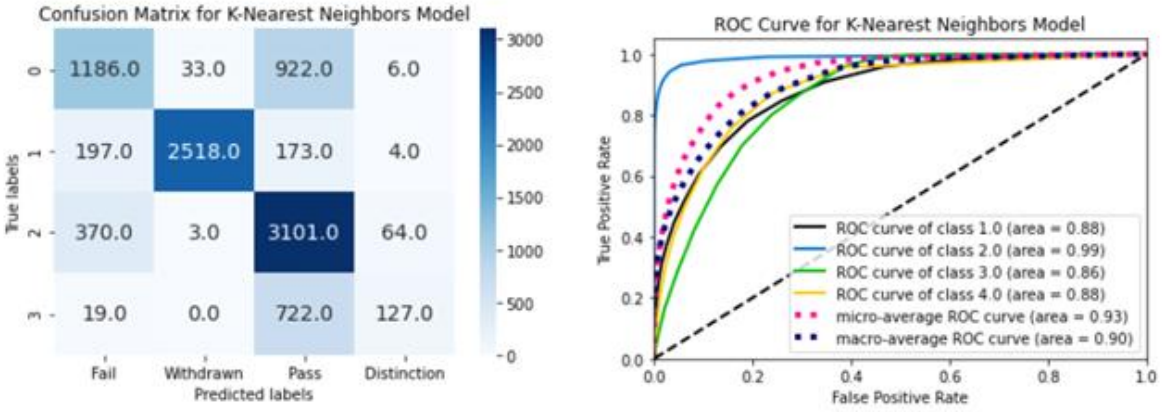


Figure 18 Confusion Matrix and ROC Curve for KNN Model

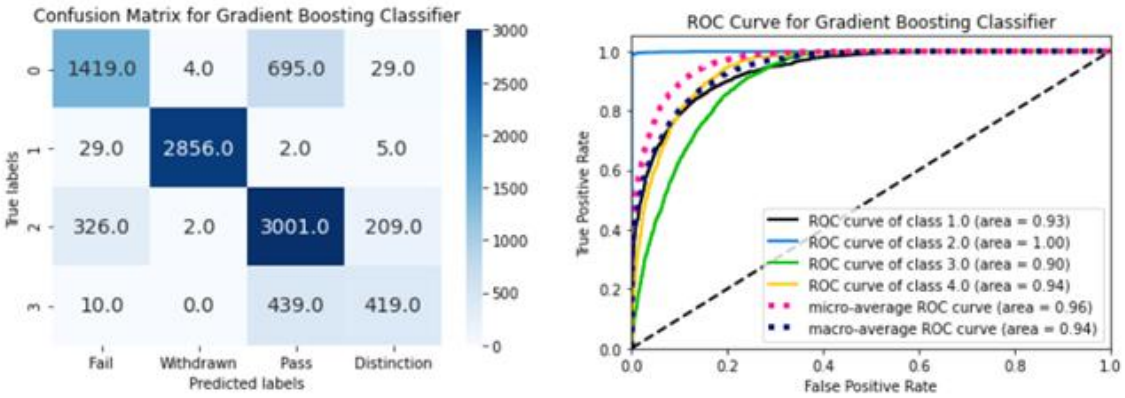


Figure 19 Confusion Matrix and ROC Curve for Gradient Boosting Classifier

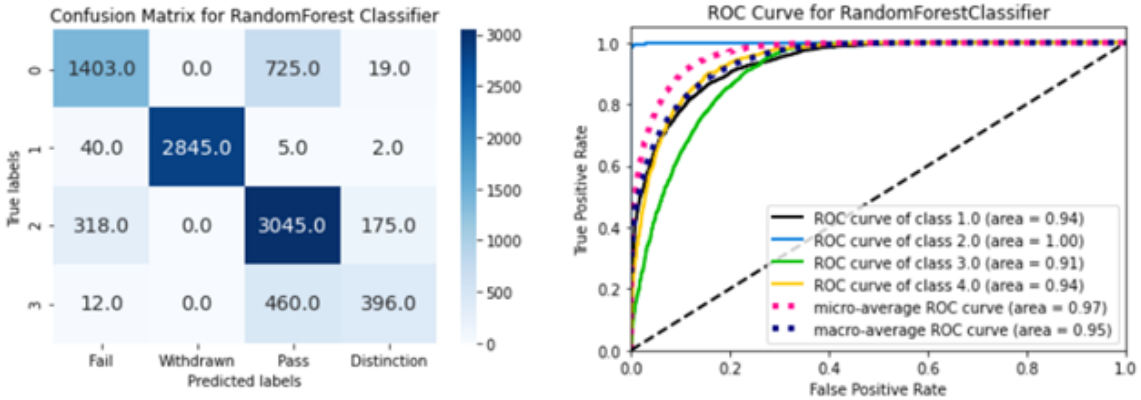


Figure 20 Confusion Matrix and ROC curve for Random Forest model

From the confusion matrix in Figure 18, Figure 19 and Figure 20 for the machine learning models namely KNN, Gradient Boosting and Random Forest models, the student final result as Fail, Withdrawn and Distinction was best predicted by Gradient Boosting model followed by Random Forest model which was slightly lesser than Gradient Boosting model. However, the final result class as Pass was best predicted by KNN model followed by Random Forest and Gradient Boosting model.

The Figure 21 shows the model plot for Multi Layer Perceptron model implemented for the OULAD dataset.

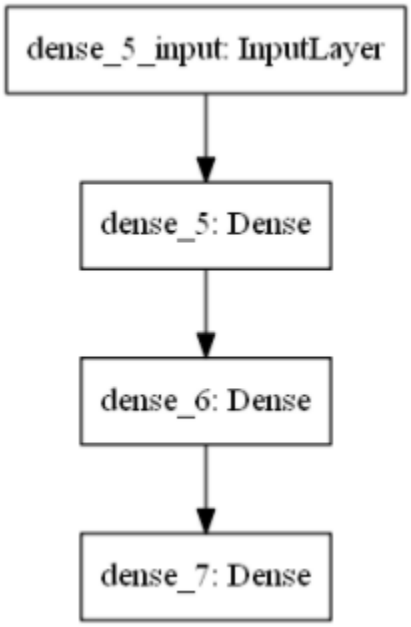


Figure 21 Model Plot for MultiLayer Perceptron

From the results above, the Multi Layer Perceptron had the highest accuracy of 90%. The next best models were Gradient Boost and Random Forest model with accuracy of 81.5%. The K Nearest Neighbour had the least accuracy of 73%.

CONCLUSION AND FUTURE WORK

The OULAD dataset is helpful for the online education training providers in identifying the students who would be able to successfully complete the registered courses. And for those who could withdraw/dropout or fail the final exam based on the suitable predictions, necessary intervention and extra follow up can be carried out for supporting the students who require additional support. Therefore, prediction of the student performance or drop out predictions through the predictive machine learning and data mining models from the OULAD or similar MOOC dataset could be advantageous to the online education providers in formalizing the course structure and taking necessary actions for maximising the learning for the students.

From the experimental results on the dataset, the Multilayer perceptron and gradient boosting algorithm are the best suited models for predicting the student performance results. The multilayer perceptron model performance could be further improved by applying hyperparameter tuning method. For future scope, the dataset could also be utilized in building LSTM and Convolutional neural network models and validating the student performance with the student clickstream and demographics data over the course of time.

REFERENCES:

[1] OULAD MOOC Dropout and Result Prediction using Ensemble, Deep Learning and Regression Techniques, Nikhil Indrashekhar Jha , Ioana Ghergulescu and Arghir-Nicolae Moldovan b School of Computing, National College of Ireland, Dublin, Ireland Adaptemy, Dublin, Ireland

[2] Predicting At-Risk Students Using Clickstream Data in the Virtual Learning Environment. Naif Radi Aljohani , Ayman Fayoumi and Saeed-Ul Hassan , 17th December 2019, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia; nraljohani@kau.edu.sa (N.R.A.); afayoumi@kau.edu.sa (A.F.) 2 Department of Computer Science, Information Technology University, Lahore 54600, Pakistan,

[3] Machine Learning Techniques for Determining Students' Academic Performance: A Sustainable Development Case for Engineering Education. Sujan Poudyal, November 2020, Conference: 2020 International Conference on Decision Aid Sciences and Application (DASA)

[4] A Deep Model for Predicting Online Course Performance Hamid Karimi, Jiangtao Huang2, Tyler Derr, Data Science and Engineering Lab, Michigan state University, {karimiha, derrtyle}@msu.edu , School of Computer and Information Engineering, Nanning Normal University, China, hjt@gxtc.edu.cn

[5] Detecting At-Risk Students With Early Interventions Using Machine Learning Techniques , RAGHAD AL-SHABANDAR, ABIR JAAFAR HUSSAIN , (Member, IEEE),PANOS LIATSIS, (Senior Member, IEEE), AND ROBERT KEIGHT

[6] Online At-Risk Student Identification using RNN-GRU Joint Neural Networks, by Yanbai He ,Rui Chen ,Xinya Li ,Chuanyan Hao ,Sijiang Liu ,Gangyao Zhang , and Bo Jiang,9th october 2020, School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China, School of Overseas Education, Nanjing University of Posts and Telecommunications, Nanjing 210023, China, School of Educational Science and Technology, Nanjing University of Posts and Telecommunications, Nanjing 210023, China

[7] Predictive Algorithms in Learning Analytics and their Fairness, Shirin Riazy1 and Katharina Simbeck2, Niels Pinkwart, Johannes Konert (Hrsg.): Die 17. Fachtagung Bildungstechnologien, Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn ,2019, 223

[8] SEPN: a sequential engagement based academic performance prediction model.", Li, Jianxin et al., Jul 8 2020,IEEE Intelligent Systems, Volume 36, Issue 1, Jan-Feb. 1 2021, Pages 46-53

[9] OU Analyse: analysing at-risk students at The Open University." Learning Analytics Review (2015): 1-16., Kuzilek, Jakub, Hlosta, Martin; Herrmannova, Drahomira; Zdrahal, Zdenek; Vaclavek, Jonas and Wolff, Annika (2015). OU Analyse: analysing at-risk students at The Open University. *Learning Analytics Review*, LAK15-1 pp. 1–16.
URL: http://www.laceproject.eu/learning-analyticsreview...

[10] Hlosta, Martin; Zdrahal, Zdenek and Zendulka, Jaroslav (2017). Ouroboros: early identification of at-risk students without models based on legacy data. In: LAK17 - Seventh International Learning Analytics & Knowledge Conference, 13-17 Mar 2017, Vancouver, BC, Canada, pp. 6–15.

[11] Open University Learning Analytics dataset | Scientific Data (nature.com) , Jakub Kuzilek, Martin Hlosta & Zdenek Zdrahal ,28th November 2017, *Scientific Data* volume 4, Article number: 170171 (2017)

[12] Santos, J. L.; Klerkx, J.; Duval, E.; Gago, D.; and Rodrguez, L. 2014. Success, activity and drop-outs in moocs an exploratory study on the UNED COMA courses. In Proceedings of the Fourth International Conference on Learning Analytics and Knowledge, 98–102. ACM

[13] Jha, N.; Ghergulescu, I.; and Moldovan, A.-N. 2019. Oulad mooc dropout and result prediction using ensemble, deep learning and regression techniques, CSEDU 2019, Computer Science

[14] Jordan, K. June 2015. Massive open online course completion rates revisited: Assessment, length and attrition. International Review of Research in Open and Distance Learning 16(3). 341-358

[15] Wikipedia contributors. (2021, May 6). K-nearest neighbors algorithm. In *Wikipedia, The Free Encyclopedia*. Retrieved 15:55, May 22, 2021, from https://en.wikipedia.org/w/index.php?title=K-nearest_neighbors_algorithm&oldid=1021698330

[16] Wikipedia contributors. (2021, May 6). Random forest. In *Wikipedia, The Free Encyclopedia*. Retrieved 15:56, May 22, 2021, from https://en.wikipedia.org/w/index.php?title=Random_forest&oldid=1021839899

[17] Wikipedia contributors. (2021, May 6). Gradient boosting. In *Wikipedia, The Free Encyclopedia*. Retrieved 15:56, May 22, 2021, from https://en.wikipedia.org/w/index.php?title=Gradient_boosting&oldid=1021757982

[18] Jason Brownlee,July 24, 2019 in Deep Learning, Your First Deep Learning Project in Python with Keras Step-By-Step, https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/

[19] Jason Brownlee, November 9, 2018 in Deep Learning for Time Series, How to Develop Multilayer Perceptron Models for Time Series Forecasting, https://machinelearningmastery.com/how-to-develop-multilayer-perceptron-models-for-time-series-forecasting/

[20] Jason Brownlee, July 21, 2016 in Deep Learning for Time Series, Time Series Prediction with LSTM Recurrent Neural Networks in Python with Keras, https://machinelearningmastery.com/time-series-prediction-lstm-recurrent-neural-networks-python-keras/

[21] Sidath Asiri, Jun 11 2018, Machine Learning Classifiers, Towards Data Science, Machine Learning Classifiers. What is classification? | by Sidath Asiri | Towards Data Science

[22] Wikipedia contributors. (2021, May 17). Precision and recall. In *Wikipedia, The Free Encyclopedia*. Retrieved 16:03, May 22, 2021, from https://en.wikipedia.org/w/index.php?title=Precision_and_recall&oldid=1023690484

[23] Jason Brownlee, November 18, 2016 in Code Algorithms from Scratch, What is Confusion Matrix in Machine Learning, https://machinelearningmastery.com/confusion-matrix-machine-learning/#:~:text=A%20confusion%20matrix%20is%20a,two%20classes%20in%20your%20dataset

[24] Jason Brownlee, February 27, 2019 in Deep Learning Performance, How to use Learning Curves to Diagnose Machine Learning Model Performance, How to use Learning Curves to Diagnose Machine Learning Model Performance (machinelearningmastery.com)