

Multi-class classification to track students' academic outcome

MSc Research Project
Data Analytics

Apurva Jain
Student ID: x18104142

School of Computing
National College of Ireland

Supervisor: Dr. Muhammad Iqbal

National College of Ireland
Project Submission Sheet
School of Computing



Student Name:	Apurva Jain
Student ID:	x18104142
Programme:	MSc. Data Analytics
Year:	2018-2019
Module:	Research Project
Supervisor:	Dr. Muhammad Iqbal
Submission Due Date:	12-08-2019
Project Title:	Multi-class classification to track students' academic outcome
Word Count:	8162
Page Count:	24

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	APURVA JAIN
Date:	12-08-2019

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission , to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project , both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Multi-class classification to track students' academic outcome

Apurva Jain
x18104142

12-08-2019

Abstract

Despite various measures taken by the universities and colleges, the number of college and university dropouts is still on the rise. About 19.5% of students in the U.K drop out of their colleges every year¹. Thus, the identification of tentative dropouts and failure-prone students can act as an early warning system while also assisting teachers in analysing the need to streamline their course according to the weaker students' need. Learners' academic performance majorly depend upon their demographic factors as well as some other learning-based factors.

Objective: This paper focuses on classifying students as Pass, Fail or Dropout, based on their demographic and learning features. Learner's demographic factors like their like Age, Gender, Highest Qualification, Index of Multiple Deficiency, Physical Disability and learning factors like student's past assessment grades, their interaction with Virtual Learning Environment have been considered for this study.

Dataset: A publicly available anonymised student data from Open University named as Open University Learning Analytics Dataset OULAD² has been used for this study.

Methodology: To carry out this multi-class classification, five different machine learning and deep learning algorithms, Artificial Neural Networks (ANN), Random Forest, Decision Trees, XGBoost, Support Vector Machine (SVM), have been implemented.

Results: Accuracy has been considered as a metric of evaluation for the models. ANN performed the best with 78.08% accuracy.

Keywords: Learning Analytics, Educational Data Mining, e-learning, multi-class classification, VLE, Deep Learning, Artificial Neural Networks

1 Introduction

Numerous measures are being taken up by educational institutions to keep a track of student's learning activities focussing on their academic results. Many latest technologies including Virtual Learning Environment (VLE) and Learning Management Systems (LMS) like Moodle, Blackboard have eased the teaching efforts. The versatility of these

¹<https://www.theguardian.com/education/2018/mar/08/university-drop-out-rates-uk-rise-third-year>

²<https://doi.org/10.6084/m9.figshare.5081998.v1>

systems have not only helped teachers but have also assisted students in keeping themselves in line with their academics. VLEs and LMS are internet-driven technological platforms which help the instructors and learners in engaging with the course content, ask out queries in forums, with another bunch of activities like quizzes, tests, assignments. Using these systems helps students due to their access anywhere anytime facility. The efficiency of these techniques can be seen in the form of decreased number of failure rate. Dropping out of college and the low academic success rate is persistent in colleges and universities as well as on internet-driven learning platforms like Massive Open Online Courses (MOOCs). Such low performances not only decrease the morale of the students but also result in wastage of precious teaching efforts. Not only this, in the long run, those who drop out are more susceptible to monetary and health issues (Christle et al. (2007)).

Thus, an effective solution to control and monitor this problem of drop out and failure is to predict in advance whether a learner is more prone to fall in either of the two categories. Learning Analytics has fairly assisted in achieving this goal but a clearer and more accurate solution needs to be defined so that identification of dropout prone and failure-prone students can be done before their final exams (Romero et al. (2008)). The number of features affecting the student drop out is still unknown and needs to be revisited. According to Stratton et al. (2007), student's factors like their age, gender, ethnicity as well as past academic results can influence their final learning outcome. Classification techniques which make use of multiple socio-economic factors can effectively predict the learning outcome in comparison to those which are not inclusive of any of these factors.

This paper presents an academic outcome prediction model which identifies the students who are on the verge of dropout or failure. Student's demography usually comprises of age, gender, the region of residence, whether they have a physical disability or not while learning factors may include a student's interactivity with their VLE as well as their previous academic learning outcome. The studies involving marks prediction or final learning outcome have been massively successful, however, these were solely based on previous marks. Hu et al. (2017a) states that these models do not capture the evolution of student's learning. Thus, considering and including all other demographics related parameters can be more insightful than the traditional approach. Work done by Ahmed & Elaraby (2014), Agudo-Peregrina et al. (2014), Jiang et al. (2014) showcase the use of traditional classification models like logistics regression, decision trees, logistic regression to predict grades with commendable results. These algorithms have optimal performance for smaller datasets but they are likely to slow down when the input data increase.

The huge amount of data generated in the form of student activity from the VLE, their personal demographic information, can be judiciously used to create models which can effectively make use of the data and bring forth some results which can be further used in the welfare of students in terms of their performance and personal growth. This requirement can be fulfilled by using deep learning techniques which are meant to process big datasets easily. Such models also provide amazing results with commendable performance in comparison to other traditional machine learning approaches. These algorithms have facilitated the processing of datasets which were initially of no use because of unavailability of powerful processors and GPUs. Adding on to their numerous advantages, Neural Nets work as non-linear classifiers wherein they can be used for data where even a significant relationship doesn't exist between dependent and independent variables, and also there, where the number of input variables is quite high, e.g. Image Classification,

Natural Language Processing (NLP) etc. (Garson (1998)). Their application in the field of grade prediction and learner’s performance prediction has brought great results (Musso et al. (2013), (Hu & Rangwala (2019), Okubo et al. (2018), Piech et al. (2015), Sivasakthi (2017)).

Hence, considering all the previous researches and past results, this paper experiments with five different multi-class classification models with a mix of traditional machine learning and state of the art deep learning techniques. Artificial Neural Network (ANN), Extreme Gradient Boosting (XGBoost), Random Forest, Decision Trees and Support Vector Machine (SVM) have been developed to classify learner’s academic outcome for the course enrolled, using their demographic as well as learning features. Students are categorised into three categories “Pass”, “Fail” or “Dropout”. Dataset for this research has been taken from Open University - OULAD³ dataset. It is an open-source, anonymised student data which lists student-related demographic as well as academic information.

The results from these models can profoundly help instructors to upgrade their existing teaching techniques and bring in some new approaches to improve students’ learning and their interests towards the enrolled course. These systems can also work as an early warning system to the scholars as well as the teacher to pay extra attention to their studies and to further accentuate their grades. Weaker and unprivileged students who tend to drop out because of their family problem can also be benefited from these models. Slow learners and drop out prone individuals can be watched over by the institutes to improve their learning process.

Further, section 2 gives the Literature Survey, section 3 presents the Proposed Solution, section 4 portrays Experimentation and section 6 and 7 have Discussion and Conclusion & Future Work respectively.

2 Related Work

To carry out this analysis, several related pieces of research have been reviewed dealing with student’s performance prediction which is discussed as follows.

2.1 Attribution for student’s performance prediction

2.1.1 Grade prediction using previous grades and demographic characteristics

An analytical model developed by Barber & Sharkey (2012) predicts the failure rate among the students of the University of Phoenix. For this study, various factors related to a student like their high school grades, their ethnicity, gender, grades in previous courses, age, financial status were considered. The model predicts the at-risk students and further prompts to take appropriate action by prioritising focus on weaker students. Two traditional machine learning models- Naïve Bayes and Logistic Regression with varying performance parameters were implemented. This included dropping of insignificant input variables and including highly significant ones.

Another grade prediction model was developed by Marbouti et al. (2016) to identify weak students at the early start of coursework. Seven different models were tested but Naïve Bayes and an Ensemble model with a mix of SVM, K-Nearest Neighbours (KNN)

³<https://doi.org/10.6084/m9.figshare.5081998.v1>

and Naïve Bayes turned out to be the best-performing ones. These model were tested on a dataset which has a standard grading approach. The author argues that all the warning systems cannot be generalised. Hence, this model was built using fourteen different input variables related to a student's in-class engagement and performance. The weekly grades of students were standardised and attribution was done using correlations. The model was designed keeping in mind the course needs. The shortcoming of such a model is that these are course-specific and cannot be used in general for other courses. Sivasakthi (2017) came up with an idea of using a student's demographics as well as previous grade in language C to predict the students who lagged in programming. The idea behind this model is great but just judging the student's performance based on their previous grades and personal details isn't enough. Again, the correlation between these variables were found which shows that the previous marks in C had a high correlation with their grades. Deep learning algorithms along with few other traditional data mining algorithms were applied where Multi-Layer Perceptron (MLP) gave the best accuracy.

Thiele et al. (2016) affirms through their research that a strong correlation exists between a person's demographic details and their respective academic performance. The author argues that the students belonging to deprived sections of society are usually underrepresented. Thus, including the demographic details in grade prediction can be more inclusive and can provide more context. Correlations were used to validate this assumption. The results identify that all the contextual features like age, ethnicity, IMD as well as school grades are significant contributors to academic performance. This study also puts forth the fact that a student's school grades do not cover the wholesome aspect of the personality of a student. It doesn't give the true potential of a learner. Thus there is a need to include all the factors which contribute to the overall learning behaviour of a student. Hu et al. (2017b) developed a hybrid model based on regression. This model uses content and context features for learning outcome prediction in a course. The study involved students' details from two different educational institutes. The model's outcome shows that content features (like high school grade), is not enough for overall performance prediction for students. Context features like sexuality, gender, etc. must also be included for better results. Running multiple test iterations revealed that both types of input features must be used to create a final input feature set. The results of the study showed that prediction models cannot be generalised. Also, the students belonging to different institutes may have different factors affecting their grades based on the layout of the curriculum. This study confirms that only marks scored in the exams are not a basis to portray the true potential of a student. The effect of non-academic features on learner's performance was tested on student's data which consists of 37 variables which constituted of details like their social background, their interactivity, their motivation levels, financial status, demography. The test was conducted on 103 different students' data by Dharmawan et al. (2018). The model results show that social contextual features play a major role in deciding the academic learning outcome of students. Decision tree-based model was developed for the whole study while considering all the significant factors for performance prediction.

The work carried by different authors in the above section demonstrates that academic performance highly relates to social demography along with previous grades. Thus, early warning systems or prediction models must include such factors for better results and optimal coverage.

2.1.2 Grade prediction using VLE interactivity and academic grades

Peach et al. (2019) used an unsupervised learning approach to outline the online learning style amongst scholars. “Markov Stability Graphs” which uses time series analysis for unsupervised learning approach was employed on 81 students’ information who were associated with six different courses. This approach helped in identifying groups of students having similar online learning behaviour. It also finds out those learner’s who are weak and the ones who are not actively seeking participation in their courses. Authors developed a learner’s similarity matrix to match and identify the ones falling in a similar category. This study helps in finding out students who tend to finish their assignments and submits them well before time, the ones who submit just on time or those who are like crammers. Once the clusters were identified, the author applied SVM and Decision Trees (DT) to identify the students based on their performances. SVM was chosen to identify pure classes of students with low, medium and high intensity of performance in final assessments, while DTs were used to group similar students. DTs were limited to 4 branches, beyond which accuracy and computational speed decreased while increasing complexity. The shortcoming of this study is the dataset used here, it is quite small and is under-representative of the students which may lead to misleading results.

Iterative experimentation was done by Agudo-Peregrina et al. (2014) used VLE interactivity of students to predict their final assessment’s performance. Different levels of interactions were taken into account. One based on student-student interaction in the form of queries posted on forums and comments, student-content interaction in the form of accessing the module content, student-teacher interaction and student-system interaction. These interactivity levels were later divided into three groups low, medium or high level of interaction. Mode of interaction with the LMS and VLE was also recorded. Thus, the final feature vector consisted of all the interactivity levels, their modes, which were later used to predict their final results. Correlations were used to find out the relationship between variables, but these results sheerly based on correlations can be error-prone. So to balance out the Multiple Logistic Regression model was also implemented. This study concluded on the note of considering more generalised models rather than being course-specific.

Elbadrawy et al. (2014) also worked on developing a multi regression model to relate students’ tentative grade with different Moodle-based activities. A linear function was used to combine their past performance data with the moodle activity logs. The model outputs the tendency of a student to pass the final examination. This model was a great work however, it lacked the demography related features which as per previous researches (Hu et al. (2017b), Barber & Sharkey (2012)) have been considered significant in predicting final grades. The dataset was obtained from real-time students studying at the University of Minnesota. This study involved tailor-made prediction for every student which is more convenient than the generalised models. The feature set included all the student’s course-related details, department details, their interaction and activity logs on Moodle like comments on forums, posts, and quiz taking behaviours. Models were developed by including and without including the Moodle interactivity and its derived features. The test results interestingly show that error values decrease on including Moodle related features. Conijn et al. (2017) analysed 17 different course modules offered to 4989 students by a University which uses Moodle LMS for their students. The author has explained the use of multi-level regression for predicting the performance of students using the features offered by LMS like quiz views, attempts per quiz, online activity,

time since the last login, etc. The author also comments on the inconsistent findings of various researches solely based on correlations. Thus to make a full proof model, a set of content usage related and content view related variables were considered. The results were validated by carrying out multiple linear correlations. This study shows that LMS related features can be highly useful in prediction student's performance.

A multi regression model to predict the learner's grades in Massive Open Online Courses (MOOC) was published by Ren et al. (2016) which is based on the work carried out by Elbadrawy et al. (2014) where student's performance was predicted using Moodle logs. Feature set included six different variables related to interactivity on MOOC. The features included homework, quizzes, videos related parameters. The personalised linear multi regression model aimed at identifying assessments results on a MOOC derived an accuracy and precision score of about 76.5% and 0.852 respectively than other baselined models.

All these models based on multiple regression clearly defines the dependability of final assessment grades on demography, academic and VLE activity-related features.

2.2 Techniques used for learning outcome prediction

2.2.1 Traditional machine learning approaches for grade prediction

A model based on C4.5 Decision Trees (DT) was implemented by Elgamal (2013) to predict the performance of students in a computer programming course. Feature selection was carried out using fuzzification of continuous variables into singular linguistic variables. Decision Tree is usually the first choice among the researchers because of their excellent capabilities and prediction powers, but, these also suffer from model overfitting which results in a huge difference between the training and testing accuracy. These models also get degraded in terms of performance when handling bigger datasets. Thus, alternative approaches like deep learning techniques can be a better replacement. Scalability is a problem encountered in classification models. To overcome this issue, Pandey & Taruna (2016) came up with an ensemble technique which consisted of three different modelling units namely DTs, KNN, and Aggregating One-Dependence Estimators (AODE). All these classifiers were aggregated into a single ensemble model using the product probability rule. All the three algorithms are powerful in itself but when combined, the new model works based on the voting system. This model turned out to be an effective approach to student classification based on socio-economic status and historic grades. This model was trained and tested on three different datasets to maximize performance. The hybrid ensemble approach imparted best results (87% accuracy) in comparison to other single classification models. A conclusion can be made based on this study that a single classifier cannot effectively predict student classes.

The study carried out by Iqbal et al. (2017) uses Matrix Factorisation & Collaborative Filtering on Electrical Engineering student's data who were enrolled at Information Technology University. A matrix was developed to find out a relationship (if exists) between admission based factors like previous CGPA, initial enter level exam marks, and marks in previous grades like senior secondary and secondary marks. The simplicity of these models eases the training and testing process for smaller data, but, it will not work for a big chunk of scattered data. A program to provide teachers with student's performance feedback was also developed in the same case study. This system uses predicted GPA of course enrolled, their domain knowledge and their knowledge inference levels in the

enrolled course to help the teacher provide special assistance to weaker students whose knowledge score comes out to be lesser than 2.67.

Another logistic regression model was proposed by Baars et al. (2017) to predict failure among first-year students of medical college. Data of 1819 students were gathered who were enrolled in Erasmus Medical School for five cohorts. Many features like demographic details including gender, age, marks obtained, their participation rate were taken as inputs which were used to predict the rate of success of students in their exams. Only statistically significant variables were chosen as input for all five cohorts. These predictions were run for five iterations after every 2 months to take timely know-how of student's performance. Logistic regression is a convenient model as the dataset used was quite small in size and the predictions are completely based on correlations. The results of each iteration were collectively looked at and it was concluded that those who managed to pass in the initial assessments were also able to finish their course on time without any fail. However, a more diverse dataset needs to be used for training and testing so that actual reality can be simulated. Smaller datasets do not completely cover all aspects of students personality, which may lead to inappropriate results. On the other hand, the results from these models can help monitor the performance of the students over the whole academic year.

Traditional machine learning algorithms like Logistic Regression, Multiple Regression, Naïve Bayes, Decision Trees have been a popular choice amongst researchers. These algorithms work fairly well when the datasets are smaller in size but more computationally competent models are required to process larger datasets.

2.2.2 Student performance prediction using deep learning techniques

Shahiri et al. (2015) identified the problem of making use of large educational data set in grade prediction. Thus, a Neural Network based model was deployed to overcome this. This study not only focuses on grade prediction but also works towards finding those attributes which significantly affect their final grades. It showcased all the major machine learning techniques used for student's performance prediction. Research reviewed by Shahiri et al. (2015) concluded that DTs, ANNs, KNN, Naïve Bayes and SVM are the most popular algorithms used for performance prediction. But amongst all Neural Networks have been the best in terms of performance. They are capable of imputing actual human brain's functionality, powerful enough to handle non-linear data as well.

A similar thorough analysis of various works in the field of learning analytics was carried out by Sivasakthi (2017). The comparative study included 5 different machine learning algorithms for supervised learning used in educational data mining. The student was performed on 300 student's information for the identification of novice programmers enrolled in the course. The purpose was to find out and help such students who might lag in their curriculum. Many features including their demographic as well as grades in the language C has used criteria to identify the weak learners. WEKA tool was used in the model development. The study results prove a neural network-based model to be the most accurate one without 93% of accuracy. The test included 10 fold cross-validation to validate the authenticity of the results obtained. Dropout Prediction System developed by Ortigosa et al. (2019) was made to process completely on real-time student data of about 11000 students which was gathered over a period of 5 years. The study focused on students retention at the university level. This work used C5.0 as their baseline algorithm for performance prediction and was able to replicate the experimental results on their real-time data. A series of event like Extract, Transform, Load (ETL) system

was configured to gather and denormalized huge amount of student information and other interaction data from LMS. On similar lines, our current work included denormalization of huge student data to carry out further analysis.

To practically find out the relationship between various features amongst various data variables in a given student dataset, Saarela & Kärkkäinen (2015) used MLP to predict the final tentative grades of students in a bachelor's course. The authenticity of this model was verified using multiple tests which confirmed that students' general studying behaviour highly affect their performance and success rate in exams, rather than their core technical skills. First, a Bonferroni correlation analysis was carried out, followed by a clustering approach to find out clusters of students with the same learning pattern. Thirdly a predictive analysis was done using previous courses those students have passed. MLP was used for this method. Comparative analysis of various machine learning techniques was done by Ibrahim & Rusli (2007) to find out the best algorithm for learner's performance prediction. This piece of work implies that the Neural Network has been the best candidate for performance prediction. Conventional approaches fail to deliver the best possible accuracy with an increase in data.

The extensive literature review clearly shows that several factors must be considered before creating the final feature set for student's performance prediction. All demographic factors, grades, and interactivity with LMS must be included in the feature set. Also, the versatility of a generalised model can make it available to users across different universities and educational institutions. None of the researchers have collectively analysed Dropout and Failure using all the potential major factors. Thus, this novel piece of work focusses on a multi-class classification model which classifies students into three categories- "Pass", "Fail", and "Dropout".

The purpose of this research is to find out the answer to the **Research Question:** "How effectively can various machine learning algorithms predict student's learning outcome by carrying out multi-class classification based on their demographics, academic, and LMS interactivity based features?"

3 Research Methodology

Data mining based research work involves complex processing and decision-making. Following a framework which outlines the set of steps and protocols can ease the entire research process. These analytical models not only help in laying out the approaches to deal with data mining projects but also smoothens the documentation process and help in gaining industry level standards, thus producing better quality work. These models have a set of phases according to which each research work progresses, ultimately making the work more proficient. There is a pool of such process models available to choose from. CRISP-DM (Cross Industry Standard Process for Data Mining), KDD (Knowledge Discovery Database), SEMMA (Sample, Explore, Modify, Model, Assess) are a few of the most popular ones. So, a critical analysis of what aligns with the research work needs to be done beforehand. Cross Industry Standard Process for Data Mining (CRISP-DM) methodology is one of the most widely used data mining process model. It's highly generic, easy to follow, has well-defined phases and its versatility in terms of future re-work makes it the most compatible approach to any data mining based project. Therefore, the same process has also been followed for student's learning outcome prediction. Many studies involving student attrition prediction have also used the

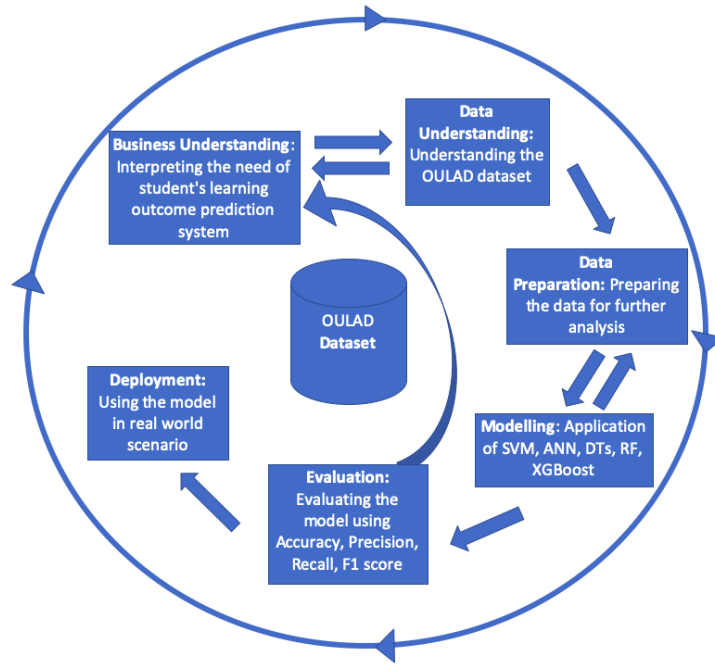


Figure 1: CRISP-DM architecture (Wirth & Hipp (2000))

same data mining methodology (Castro R. et al. (2018), Mariscal et al. (2010), Wirth & Hipp (2000), Castro R. et al. (2018)). CRISP-DM comprises of six main phases which are Business Understanding, Data Understanding, Data Preparation, Data Modelling, Evaluation and Deployment, as shown in figure 1.

All the six phases of CRISP-DM concerning our current research work have been explained below in the following subsections.

3.1 Business Understanding

Business Understanding is the first and the most crucial phase of CRISP-DM methodology. It is aimed at getting hold of complete knowledge of the research domain in terms of business perspective. It involves thorough analysis and understanding of the research objective and penning down the research question. Research goals are set up and are clearly defined in this phase giving a proper understanding of the Business aspect of the research. Business Understanding consists of three major sub-parts namely Business/Project Goal, Business/Project Plan and Business/Project Plan Assessment.

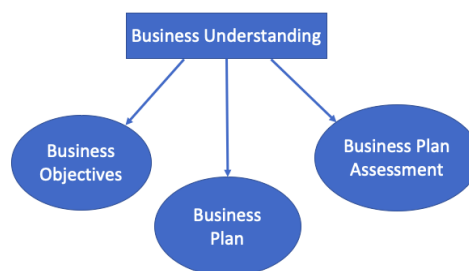


Figure 2: Business Understanding

Project Goal: The objective of this work is to predict students' tentative success, fail-

social features like Education, Health, Employment, Living Conditions, Skills, and Training, etc.

2. **studentVle:** This table has the record of each student's VLE interaction. It contains the number of clicks, date, module code, and module presentation enrolled for by the student. This data file is connected to other files of the dataset using its foreign key which is code_module. The table has about 10.65 million records of each student recorded over the span of course length.
3. **vle:** Since each student enrolled for the course uses the VLE over the complete course presentation time, this table gives the details of various sites available on the VLE for students, the course length in days. It lists all the 20 different URLs embedded on the VLE for the students to access as additional resources for their coursework.
4. **studentRegistration:** This table reflects the details about the date of registration and deregistration from the course.
5. **studentAssessment:** This table lists the date of submission of various assessments as well as their respective scores obtained by each student in each of these assessments.
6. **assessments:** It describes all the types of assessments which takes place over the course length, their assessment types like Computer-Marked Assignment (CMA), Tutor-Marked Assignment (TMA) or final exam and the date of assessment.
7. **courses:** This table lists all the 22 different courses, their date of commencement.

3.3 Data Preparation and Pre-processing

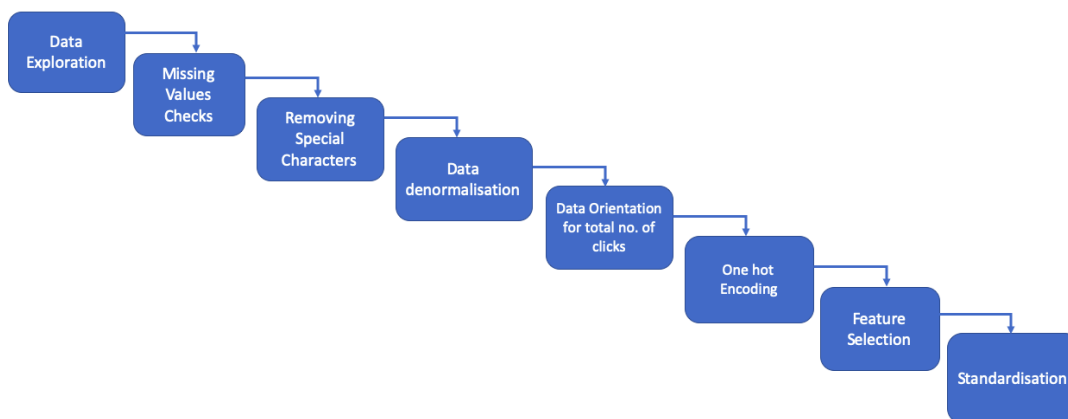


Figure 4: Data Preparation

This phase deals with preparing the data to use it for modelling. It has to be in perfect condition before it can be used further. Missing value checks, the orientation of the data as per the requirement are the few most important steps, which if missed, might bring wrong or inconsistent results. About 80% of the total data mining efforts are usually taken up in this phase. Following pre-processing steps have been taken to ensure the data is consistent concerning our data mining application.

1. **Data Exploration:** To get the understanding of distributions of a data point across all the 7 data tables, basic visualisations have been carried out using Python. This has helped understand the distribution of data across different classes. Some of the visualisations are shown in figure 5. Correlation between variables has also been calculated.
2. **Missing Values Checks:** Missing values are the roadblocks in data modelling approaches. Thus, consistency and missing value check have been implemented. Missing values and outliers are common with the large real-time dataset. Therefore, missing values have been removed. Imputation didn't seem to be a great choice because of the variety in the chosen dataset.

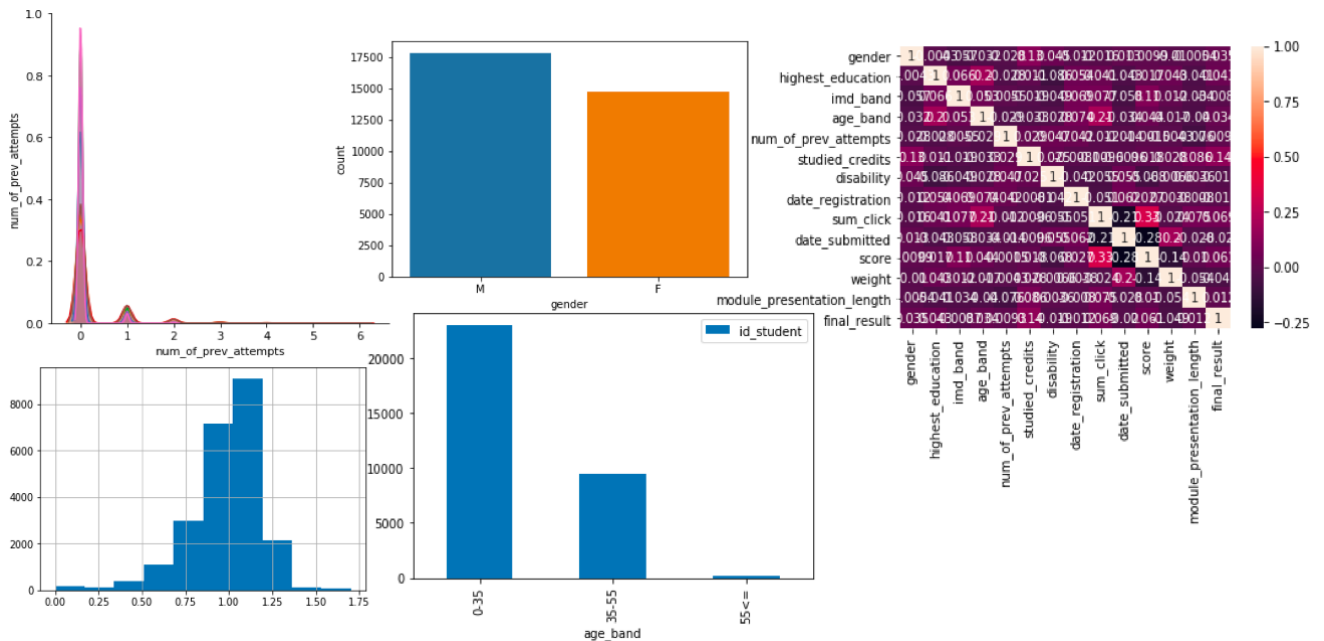


Figure 5: Data Visualisation: Distribution of total number of attempts by students in different courses(top-left), gender distribution (top-right), normal distribution of marks scored by students in various assessments (bottom-left), age distribution among students (bottom-right) and the correlation plot (right-most)

3. **Removal of Special characters:** Special and unwanted characters are equally responsible for bringing down the data quality. Thus, to prevent the model from misbehaving, all the special characters and unwanted values have been removed.
4. **Type Validation:** This is another step for data integrity. Data type validation needs to be performed before actually using it in the application. This prevents unwanted warnings and types of mismatch errors.
5. **Data Denormalization:** Since a relational dataset has been used in this study (refer figure 3), a denormalisation step is a must to merge and integrate all the seven tables in one file. This step ensures the integrity of the data and makes further analysis easy and hassle-free.
6. **Data Orientation:** Information regarding VLE interactivity of students needs to re-oriented to get the sum of all the clicks of students in the complete course length.

Thus, data orientation has been done which sums up all the clicks of each student on different sites of VLE. The output table after orientation is shown in figure 6.

7. **Data Encoding:** The dataset consists of various categorical variables. Since these features get converted to nominal and ordinal types while feature engineering, programming frameworks might consider them as integers and throw errors or may give incorrect outputs. Thus to prevent this, one hot encoding is done for all the categorical data to create dummy variables. The data after encoding is shown in figure 7.

id_student	gender	highest_education	imd_band	age_band	num_of_prestudied_cre	disability	final_result	date_registr	sum_click	date_submi	score	date	weight	module_presentation_length
559551	M	Lower Than A Level	90-100%	0-35	0	30 N	Fail	-43	770	32	84	33	16	268
559617	F	HE Qualification	20-30%	0-35	0	90 Y	Pass	-31	1104	31	83	33	16	268
559758	M	Lower Than A Level	60-70%	35-55	0	30 N	Pass	-35	518	36	61	33	16	268
560326	M	Lower Than A Level	40-50%	35-55	0	90 N	Withdrawn	-95	423	40	47	33	16	268
560502	M	Post Graduate Qualificatio	10-20%	35-55	0	30 Y	Fail	-31	1074	33	77	33	16	268
560734	M	A Level or Equivalent	70-80%	0-35	0	120 N	Fail	-28	1553	33	100	33	16	268
560876	M	Lower Than A Level	40-50%	0-35	0	60 N	Pass	-19	774	32	88	33	16	268
560928	M	A Level or Equivalent	0-10%	0-35	0	60 N	Pass	-138	1340	32	92	33	16	268

Figure 6: Student demographic and total sum clicks with the VLE for each student.

Student_ID	Gender	IMD_Band										Age_Band			HigherQualification					PreviousAttempts	Disability	Sum_VLE_Clicks				
		0-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80-90	90-100	0-35	35-55	>=55	HE Qualificatio	A Level or Equivalent	Lower Than A Level	Post Graduate Qualification	No Formal quals							
31663	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5906	
50993	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	176
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
29144	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1914

Figure 7: One hot encoded demographic data

8. **Standardisation:** Standardisation of the data variables is done to bring all the data values on the same scale. This decreases the processing time and gives more optimal results.

3.4 Modelling

To develop a model based on machine learning algorithms for student's learning outcome prediction using various socio-demographic, VLE interactivity and academic learning factors, a vast literature has been reviewed in section 2. This intensive review showcased that Deep learning-based algorithms have outperformed the conventional machine learning techniques in student attrition prediction (Shahiri et al. (2015)). Thus, a variety of conventional machine learning including ensemble approaches as well as deep learning techniques have been applied to achieve the research objective. Fulfilling the same purpose, ANN, SVM, Decision Trees, Random Forests and XGBoost have been modelled.

3.4.1 Artificial Neural Network (ANN)

Since their discovery at the beginning of the twentieth century, Neural Networks have gained immense popularity because of their excellent capabilities to impute a human brain. These have been one of the best and most versatile black box based supervised learning approach. Their application in the field of regression, prediction, pattern recognition, image recognition, Natural language processing has brought a new revolution in the domain of Artificial Intelligence. A perceptron is the most basic unit of a Neural

Network. A simple neural network is made up of several layers of these perceptrons. A Perceptron is a binary linear classifier which always either outputs +1 or -1. However, when these perceptrons are arranged together in a set of layers, they cumulatively act as a powerful non-linear classifier. A neural network comprises of several layers of perceptron arranged in a web of neurons. The first layer is termed as an input layer which receives the weighted input, the last layer is termed as output layer and with few hidden layers in between. The neurons in the input layer receive a raw input vector and get activated only when the weighted sum of all the input values is higher than the given threshold value. The threshold value is decided by an activation function. This activated input from input layer is then passed to the series of hidden layers in the same process, reaching the final output layer which gives the final result.

3.4.2 eXtreme Gradient Boost (XGBoost)

XGboost is another machine learning-based ensemble approach which uses decision trees as its base algorithm. It is an advanced version of Gradient Boosting which uses pruning, parallel processing, regularisation, and handles missing values to create a balance between over-fitting and model bias. These days XGBoost is considered as the queen of machine learning approaches. It is also one of the most popular choices for any kind of classification or prediction based data. It is comparatively faster in computation than other conventional tree-based algorithms like Decision Trees and Random Forests.

3.4.3 Decision Trees

This algorithm is based on partitioning the data into smaller and smaller chunks based on the key features. These are a simple set of logical rules which can easily be understood without any technical or statistical knowledge. These models are built in the form of tree structures, where each branch represents a decision. Customer churn analysis, credit card defaulter application are some of the major applications of Decision Trees.

3.4.4 Random Forests

Random Forests is the most widely used ensemble approach based on implementing multiple decision trees. It uses the concept of “Bagging” and “Boot-strapping”. It uses the implementation of multiple decision trees which uses different bootstrapped data on each tree. Random forest is now widely used for a classification problem like fraud detection, spam detection as well as on regression and unsupervised learning approaches.

3.4.5 Support Vector Machine

Support Vector Machine has been one of the most prominent machine learning algorithms used for binary as well as multi-class classification. This algorithm is popular because of its simplicity and excellent supervision qualities. It has been an absolute choice for basic classification to complex modern-day problems. It is the simplest of Neural Network model, based on a perceptron guided by a kernel. Nowadays SVM is implemented for pattern recognition, grade prediction, spam detection. It works work on non-linear data. Therefore, it is used as one of the baseline models for student’s learning outcome prediction.

3.5 Evaluation

This step involves evaluating the model for best possible results aligning with the research objectives. To evaluate the model Accuracy, Precision, Recall, and F1 score has been chosen as the metrics. The higher value of Accuracy implies better capability of the model to categorise the student learning outcome. True positive rate, also known as Sensitivity or Recall is the ability of the model to correctly classify positive class. F1 score is the overall model’s test of accuracy, finding the perfect balance between precision and recall.

3.6 Deployment

The final step of this cycle is to recognize the actual use of this model in the real-world application.

4 Implementation

A series of steps have been followed before the model is ready for evaluation and final reporting. Data gathering and pre-processing have been carried out as explained in section 3.2 and 3.3. The flow chart of the complete process is shown in figure 8.

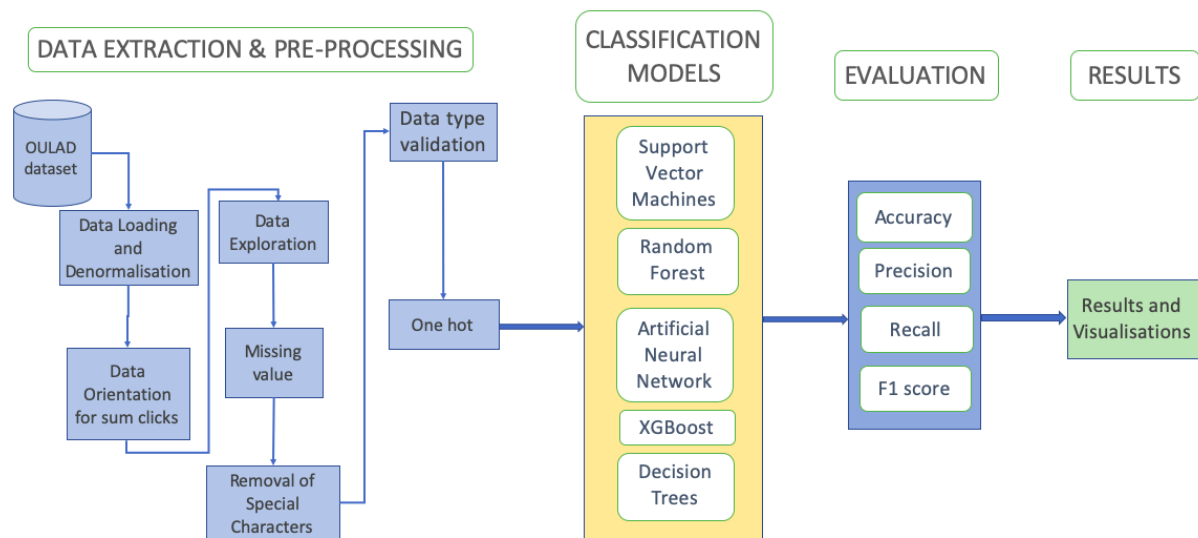


Figure 8: Data flow chart

4.1 Data preparation and pre-processing

1. **Data Gathering:** The implementation begins with data gathering which includes downloading the dataset for further pre-processing. Seven different relational data tables were downloaded in the form of .csv files.
2. **Data Loading:** As explained in section 3.2, the data varies from 10.65 million rows in studentVle table to 22 rows in courses table. Thus, to bring all this together at one place, into a single table, data denormalisation is required. To facilitate

it, a Relational Database Management System (RDMS) is used. 7 different downloaded .csv files are imported into SQL Server Management Studio (SSMS) where 7 different relational data tables are created. Since these tables are linked together using various primary and foreign keys, several SQL queries and joins are applied to merge and denormalize these data tables. This merger resulted in a single, huge, denormalised data table bearing all the student information. This step has been the most crucial and the most time consuming one. After denormalization, total sum clicks of students on different sites of VLE are calculated by aggregating all the click of each student in SQL Server using “group by” queries. This aggregated column representing the sum of clicks is added as another column in the denormalized dataset. For modelling, unique student records were required. Since data de-normalization results in repeated data values, another SQL query is run to store only the unique student’s information, resulting into a final data set ready for further analysis, exported into a .csv file.

3. **Data Exploration:** Understanding the distribution of data across all the parameters of the student data table is highly essential. This gives complete know-how of the data. Several plots are prepared in Python using its plot and seaborn libraries. Figure 5 shows some of the key visualisations including correlation matrix highlighting major correlation and other tables showing general data distribution.
4. **Missing Value checks:** Next, data cleaning is done in R. Starting with missing values, all the nulls and NAs have been removed from the data. No form of data imputation is used to maintain the authenticity of the data.
5. **Data type conversion:** All the categorical variables which are read as characters or strings are converted into factors with their respective number of levels. Incorrect assignment of data type is also checked and corrected.
6. **Feature Engineering:** Few of the columns in the dataset are unwanted like “is_banked”, “unregistration_date”. These features did not make to the final feature set, hence were removed using R packages.

This cleaned data file is then imported into Python for different machine learning models’ implementation.

7. **One-hot Encoding:** Categorical variables are one-hot encoded to prevent them from being interpreted as integers. This encoding is done in Python using Pandas libraries with the help of “get_dummies” function.

4.2 Classification Models

Many different conventional machine learning, ensemble, and deep learning-based model have been applied to the given data to find out which of the used classifiers perfectly meets the research objective. The models are run on Python 3.7 using Jupyter Notebook. All the models have been split into a 75%-25% training and testing ratio to maintain homogeneity in configuration and modelling across all the models, resulting in an easy comparison of models. Moreover, a set of 4 different datasets with varying sizes of i.e files bearing 1000, 5000, 10,000 and 22,000 rows have been created to test the performance of each of the models with changing data sizes. The input feature vector consisted of 28 variables and 3 output predicting variables for each class. The figure 9

shows the ANN implementation. Using TensorFlow CPU’s framework in backend with

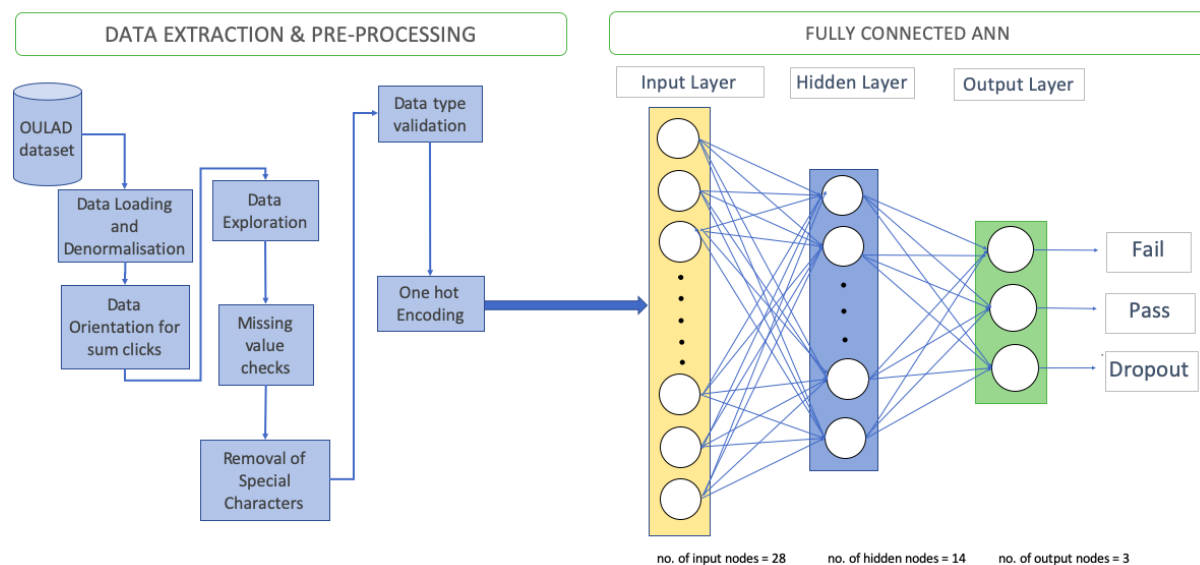


Figure 9: ANN model

the help of Python’s Keras as well as Scikit-learn libraries, ANN has been implemented. Experimentation included playing with varying numbers of epochs, hidden layers, activation functions. There were 28 input variables, so two hidden layers with 28 and 14 nodes respectively and the output layer with 3 nodes have been implemented. Batch size has been set to 16 and epochs as 100 to maintain a balance between underfitting and overfitting. Adam optimizer with multiple-cross-entropy as the loss function has been implemented. Data is standardised before feeding it to the network. Conventionally, the number of nodes in the hidden layer is taken as half the number of input variables. In this experiment, to maximize the performance, a combination of the different number of hidden layers in the neural network has been applied. The second model is XGBoost which uses xgboost library in python. Multi-class classification done on this model was optimised by specifying the maximum number of tree branches to be considered. The number of training iterations was set to 50 and an error optimisation function softprob is used. Rest of all the models have also been implemented using the scikit-learn library, with different classifier functions. The third model is RandomForest classifier which has been implemented by first standardising the input and using the RandomForestClassifier function to fit the train data. The fourth model is SVM which is implemented using SVM Classifier. A linear kernel is used while fitting the training and during the testing phase. The fifth and final model is the Decision Trees. DTs turned out to be the least performing model with the lowest performance accuracy out of all the models. The tree branches were limited to 4 to improve the model accuracy but it didn’t much affect the model’s performance as compared to the one without putting any limitations on tree branching.

5 Evaluation

Accuracy: Past research work in the field of academic outcome prediction like the works done by Hu et al. (2017a), Okubo et al. (2018) Peach et al. (2019), showcase the use of

accuracy as one of the most important performance measure. Thus, on evaluating accuracy scores for this experiment, Artificial Neural network gained the highest performance of 78.08%, followed by XGBoost with 76.11%. The accuracy values for all models are compiled in table 1. It is clear from the table that Decision Trees have been the poor performers in terms of prediction accuracy for almost the data sizes.

Classifier \ Data Size	1000	5000	10000	22000
ANN	68.54%	70.75%	72.61%	78.08%
XGBoost	68.4%	70.13%	73.04%	76.11%
SVM	59.91%	63.14%	70.9%	74.84%
Random Forest	64.8%	67.9%	68.6%	72.61%
Decision Trees	52.74%	60.19%	66.23%	71.37%

Table 1: Accuracy of implemented classifiers

Precision: This metric describes how correctly the classifier identifies the class. It is the ration of true positives identified and the total of a true positive and false positive. Since, this experiment deals with multi-class classification, the precision values for each class is calculated separately. Hence, to get an aggregate result, a micro-average parameter is included which performing the overall calculation of precision for a model. The micro-average strategy is used specifically for data sets having a class imbalance. Since the student is based on real-time application, the number of student passing any course will always be higher than the number of failures or dropouts. Thus, the micro average has been used for this dataset. It can be seen from the table 2 that ANN has been consistent and precise in the prediction of correct classes.

Classifier \ Data Size	1000	5000	10000	22000
ANN	.73	.72	.73	.81
XGBoost	.59	.57	.65	.54
SVM	.35	.32	.57	.46
Random Forest	.55	.57	.58	.55
Decision Trees	.47	.36	.40	.55

Table 2: Precision in performance of implemented classifiers

Recall: Only selecting accuracy as the metric does not give the correct overview of the model’s performance. So, Recall which is the measure of the identification of models ability to correctly classifying the true class has been used. The Recall values for all the models are listed in table 3. Again the micro average has been taken considering the class imbalance in the dataset.

F1 score: F1 scores give the overall performance of the model in terms of Recall and Precision. When there is a perfect balance between the precision and recall values, the f1 score is 1.0 which is the perfect score for a classifier. The model results for the F1 score are shown in table 4. ANN has been the most consistent model in terms of performance with the highest values. F1 score of .76 has been recorded for 22k dataset followed by decision trees and Random Forests. A number of iterations were also run for ANN using

Classifier \ Data Size	1000	5000	10000	22000
ANN	.55	.6	.63	.72
XGBoost	.54	.53	.59	.47
SVM	.55	.53	.63	.71
Random Forest	.51	.52	.55	.49
Decision Trees	.45	.47	.55	.56

Table 3: Recall values of implemented classifiers

Classifier \ Data Size	1000	5000	10000	22000
ANN	.62	.65	.67	.76
XGBoost	.55	.53	.61	.49
SVM	.38	.36	.58	.45
Random Forest	.52	.53	.55	.51
Decision Trees	.45	.40	.46	.55

Table 4: F1 scores of implemented classifiers

different epochs like 100, 200, 500 and 1000. However, the training accuracy increased but the testing accuracy decreased due to overfitting. Thus, the epochs were set to 100 for final model performance evaluation. Similarly, various hidden layers were added to optimise the model, which did not much affect the model’s behaviour. When the model was run with the hidden layer configuration of 28, 14, 14 and 3, the training accuracy came out to be 73.85 and test accuracy was as low as 68.35%. Changing the configuration to 28, 14, 7, 3 also gave about 68% of accuracy. Another addition of the hidden layer with 7 nodes also didn’t affect the accuracy. The accuracy range for neural network varied from 68 to 75%. It is often said that simple model works best, thus when a simple network with 3 layers bearing 28, 14, and 3 nodes as input, hidden and output layers respectively.

6 Results and Discussion

In this paper, a student’s final learning outcome prediction model has been prepared. The goal of this study is to develop an early warning system which predicts the final exam’s result of a student. It inputs a variety of learning, demographic and VLE interaction parameters as the independent variables to output the student’s overall result in their final exams. A combination of these input variables hasn’t yet been used collectively in predicting the student’s performance which brings novelty to this piece of work. The intensive literature survey (refer section 2) showed that Artificial Neural Networks, Decision Trees, XGboost, SVM, Random Forests are some the majorly used and most consistent algorithms in the domain of Learning Analytics (LA) and Educational Data Mining (EDM). Therefore, all these models have been implemented to the current study to find out the best performing one which works for student data. The figure 10 presents the comparison of all the implemented models in terms of accuracy and F1 score. Out of the models used as multi-class classifiers in this study, Artificial Neural Networks have

been the best-performing ones.

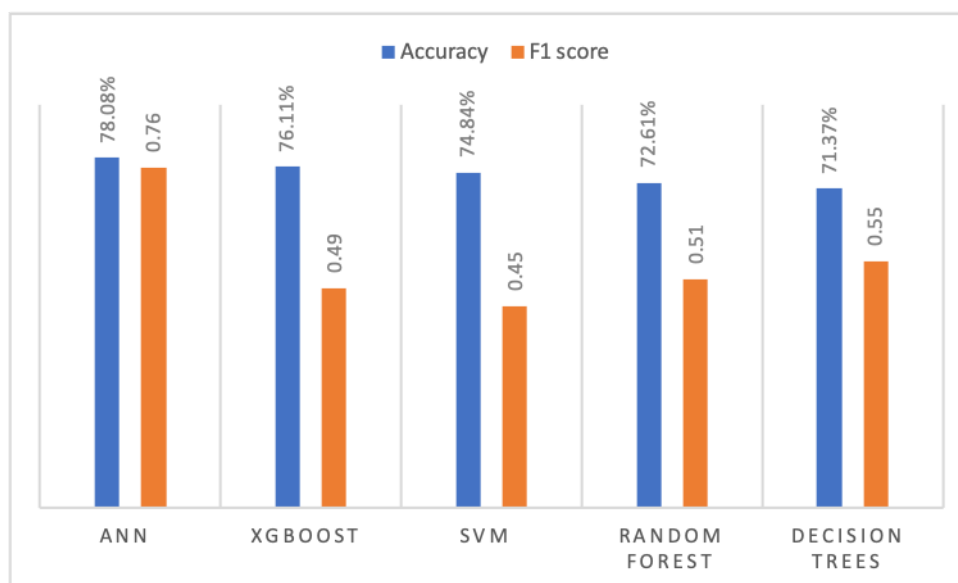


Figure 10: Accuracy and F1 score of all 5 implemented classifiers)

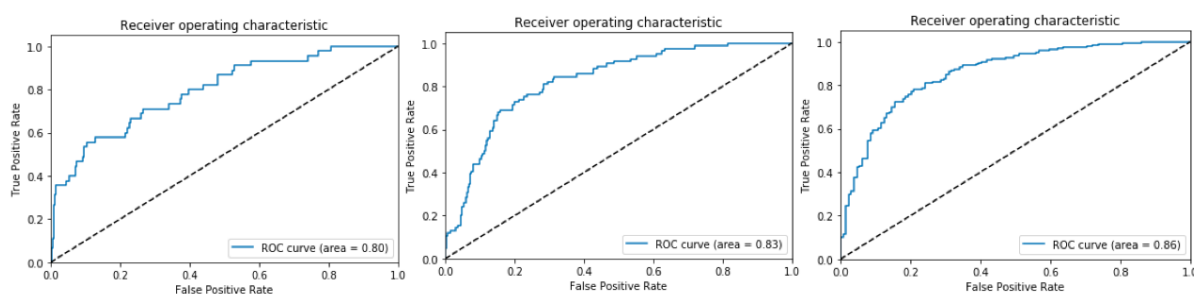


Figure 11: ROC curve for ANN model of class Dropout, Fail and Pass (left, middle and right respectively)

Student grade prediction model by Hu & Rangwala (2019) achieved F1 score of 0.38 using Multi-layer perceptron (MLP) a type of neural network similar to ANN. However, our current experiment with ANN achieves F1 score of .62 which is quite commendable. The figure 11 illustrated the ROC curve of the ANN model for all three predicted classes. The Area Under the Curve or AUC value is quite high almost near about .8 which is a good score. However, the model works best for the “Pass” category. It can also be seen that for “Dropout” class, the initial values of AUC is quite better than the rest of the two classes, but decreases with the increase in False Positive Rate. The ROC curve for “Fail” class is also quite better than the results of experimentation done by Sivasakthi (2017). The accuracy score for the model in the study by Conijn et al. (2017) is 67% in contrast with our current model which outputs about 78% score. In the experimentation setup by Dharmawan et al. (2018) non-academic data was used for dropout detection. DTs and SVM both gained 66% of accuracy while in the current setup DTs and SVM perform way better than the one implemented by the author, with about 71% and 74% of accuracy scores. The overall outcome of this study shows that the proposed prediction approach works fairly well.

7 Conclusion and Future Work

This research work focuses on implementing a students' outcome prediction model which accounts for students' demographic, academic and VLE interactivity details as input features. A series of experiments have been carried out to find out the best suitable model for such data. A handful of conventional, as well as deep learning-based classifiers, have been used to carry out multi-class classification. The results show that Artificial Neural Networks have been the best fit for this current work with the highest accuracy amongst all the other implemented supervised learning techniques. The accuracy and F1 score obtained by the model is quite commendable, however, there is always a room for improvement. Various iterations run on different data sizes resulted in the best possible accuracy. The moderation in performance could be because of the inclusion of demographic and academic features together which could be addressed in future work. Also, the data imbalance in all the three classes has added affected all models' performance. In future work, an updated version of this model can be implemented for marks prediction, wherein a student's approximate marks in the final exams can be regressed. This will give a close idea of how their final exam's marks might look like based on their current learning pace. Additionally, the inclusion of temporal characteristics of students' behaviour like weekly interactivity on different VLE sites, and timely in-class assessment results in the same model can result in a performance tracking system which notifies instructors about their students' tentative performances in timely intervals with weekly or monthly frequency.

Acknowledgement

This work has been successfully carried out with the help of my thesis supervisor Dr. Muhammad Iqbal, who has guided me throughout the coursework with his valuable inputs in each of the project meeting sessions. I pay my kind regards to my family and friends who kept my morale high and motivated me to do better. In the end, I am grateful to the creators of OULAD dataset for giving me access to their dataset.

References

- Agudo-Peregrina, Á. F., Iglesias-Pradas, S., Conde-González, M. Á. & Hernández-García, Á. (2014), 'Can we predict success from log data in vles? classification of interactions for learning analytics and their relation with performance in vle-supported f2f and online learning', *Computers in human behavior* **31**, 542–550.
- Ahmed, A. B. E. D. & Elaraby, I. S. (2014), 'Data mining: A prediction for student's performance using classification method', *World Journal of Computer Application and Technology* **2**(2), 43–47.
- Baars, G. J., Stijnen, T. & Splinter, T. A. (2017), 'A model to predict student failure in the first year of the undergraduate medical curriculum', *Health Professions Education* **3**(1), 5 – 14.
URL: <http://www.sciencedirect.com/science/article/pii/S2452301116301262>

- Barber, R. & Sharkey, M. (2012), ‘Course correction: Using analytics to predict course success’, *ACM International Conference Proceeding Series* .
- Castro R., L. F., Espitia P., E. & Montilla, A. F. (2018), Applying crisp-dm in a kdd process for the analysis of student attrition, *in* J. E. Serrano C. & J. C. Martínez-Santos, eds, ‘Advances in Computing’, Springer International Publishing, Cham, pp. 386–401.
- Christle, C. A., Jolivette, K. & Nelson, C. M. (2007), ‘School characteristics related to high school dropout rates’, *Remedial and Special education* **28**(6), 325–339.
- Conijn, R., Snijders, C., Kleingeld, A. & Matzat, U. (2017), ‘Predicting student performance from lms data: A comparison of 17 blended courses using moodle lms’, *IEEE Transactions on Learning Technologies* **10**(1), 17–29.
- Dharmawan, T., Ginardi, H. & Munif, A. (2018), Dropout detection using non-academic data, *in* ‘2018 4th International Conference on Science and Technology (ICST)’, pp. 1–4.
- Elbadrawy, A., Studham, S. & Karypis, G. (2014), ‘Personalized multi-regression models for predicting students performance in course activities’, *UMN CS* pp. 14–011.
- Elgamal, A. F. (2013), An educational data mining model for predicting student performance in programming course.
- Garson, G. D. (1998), *Neural networks: An introductory guide for social scientists*, Sage.
- Hu, Q., Polyzou, A., Karypis, G. & Rangwala, H. (2017a), Enriching course-specific regression models with content features for grade prediction, *in* ‘2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA)’, IEEE, pp. 504–513.
- Hu, Q., Polyzou, A., Karypis, G. & Rangwala, H. (2017b), Enriching course-specific regression models with content features for grade prediction, *in* ‘2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA)’, pp. 504–513.
- Hu, Q. & Rangwala, H. (2019), Reliable deep grade prediction with uncertainty estimation, *in* ‘Proceedings of the 9th International Conference on Learning Analytics & Knowledge’, LAK19, ACM, New York, NY, USA, pp. 76–85.
URL: <http://doi.acm.org/10.1145/3303772.3303802>
- Ibrahim, Z. & Rusli, D. (2007), Predicting students’ academic performance: comparing artificial neural network, decision tree and linear regression, *in* ‘21st Annual SAS Malaysia Forum’, SAS Kuala Lumpur, pp. 1–6.
- Iqbal, Z., Qadir, J., Mian, A. N. & Kamiran, F. (2017), ‘Machine learning based student grade prediction: A case study’, *arXiv preprint arXiv:1708.08744* .
- Jiang, S., Williams, A., Schenke, K., Warschauer, M. & O’dowd, D. (2014), Predicting mooc performance with week 1 behavior, *in* ‘Educational Data Mining 2014’.
- Kuzilek, J., Hlosta, M. & Zdrahal, Z. (2017), ‘Open university learning analytics dataset’, *Scientific data* **4**, 170171.

- Marbouti, F., Diefes-Dux, H. A. & Madhavan, K. (2016), ‘Models for early prediction of at-risk students in a course using standards-based grading’, *Computers Education* **103**, 1 – 15.
URL: <http://www.sciencedirect.com/science/article/pii/S0360131516301634>
- Mariscal, G., Marban, O. & Fernandez, C. (2010), ‘A survey of data mining and knowledge discovery process models and methodologies’, *The Knowledge Engineering Review* **25**(2), 137–166.
- Musso, M. F., Kyndt, E., Cascallar, E. C. & Dochy, F. (2013), ‘Predicting general academic performance and identifying the differential contribution of participating variables using artificial neural networks.’, *Frontline Learning Research* **1**(1), 42–71.
- Okubo, F., Yamashita, T., Shimada, A., Taniguchi, Y. & Konomi, S. (2018), On the prediction of students’ quiz score by recurrent neural network, *in* ‘CEUR Workshop Proceedings’, Vol. 2163.
- Ortigosa, A., Carro, R. M., Bravo-Agapito, J., Lizcano, D., Alcolea, J. J. & Blanco, (2019), ‘From lab to production: Lessons learnt and real-life challenges of an early student-dropout prevention system’, *IEEE Transactions on Learning Technologies* **12**(2), 264–277.
- Pandey, M. & Taruna, S. (2016), ‘Towards the integration of multiple classifier pertaining to the student’s performance prediction’, *Perspectives in Science* **8**, 364 – 366. Recent Trends in Engineering and Material Sciences.
URL: <http://www.sciencedirect.com/science/article/pii/S2213020916300982>
- Peach, R. L., Yaliraki, S. N., Lefevre, D. & Barahona, M. (2019), ‘Data-driven unsupervised clustering of online learner behaviour’, *arXiv preprint arXiv:1902.04047* .
- Piech, C., Spencer, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L. J. & Sohl-Dickstein, J. (2015), ‘Deep knowledge tracing’, *CoRR* **abs/1506.05908**.
URL: <http://arxiv.org/abs/1506.05908>
- Ren, Z., Rangwala, H. & Johri, A. (2016), ‘Predicting performance on mooc assessments using multi-regression models’, *arXiv preprint arXiv:1605.02269* .
- Romero, C., Ventura, S., Espejo, P. G. & Hervás, C. (2008), Data mining algorithms to classify students, *in* ‘Educational data mining 2008’.
- Saarela, M. & Kärkkäinen, T. (2015), ‘Analysing student performance using sparse data of core bachelor courses’, *Journal of educational data mining* **7**(1).
- Shahiri, A. M., Husain, W. & Rashid, N. A. (2015), ‘A review on predicting student’s performance using data mining techniques’, *Procedia Computer Science* **72**, 414 – 422. The Third Information Systems International Conference 2015.
URL: <http://www.sciencedirect.com/science/article/pii/S1877050915036182>
- Sivasakthi, M. (2017), Classification and prediction based data mining algorithms to predict students’ introductory programming performance, *in* ‘2017 International Conference on Inventive Computing and Informatics (ICICI)’, pp. 346–350.

Stratton, L. S., O'Toole, D. M. & Wetzel, J. N. (2007), 'Are the factors affecting dropout behavior related to initial enrollment intensity for college undergraduates?', *Research in Higher Education* **48**(4), 453–485.

Thiele, T., Singleton, A., Pope, D. & Stanistreet, D. (2016), 'Predicting students' academic performance based on school and socio-demographic characteristics', *Studies in Higher Education* **41**(8), 1424–1446.

Wirth, R. & Hipp, J. (2000), Crisp-dm: Towards a standard process model for data mining, in 'Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining', Citeseer, pp. 29–39.

Appendix

All the ancillary files related to the project are present at GitHub. These can be accessed via URL: https://github.com/apurvaj1204/x18104142_ThesisProject.

The OULAD dataset can be used publically and is an anonymised dataset. Refer figures 12, 13, 14 for more information.

Rights statement and License

License

This dataset is released under [CC-BY 4.0](#) license.

Citing the dataset

When citing the dataset please use the following reference:

Kuzilek J., Hlosta M., Zdrahal Z. [Open University Learning Analytics dataset](#) *Sci. Data* 4:170171 doi: 10.1038/sdata.2017.171 (2017).

Figure 12: Dataset usage permission

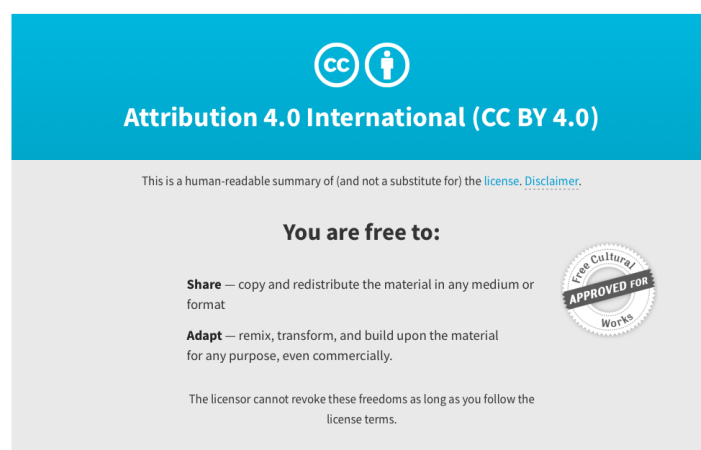



Figure 13: Dataset usage licence - part 1

Under the following terms:

 **Attribution** — You must give [appropriate credit](#), provide a link to the license, and [indicate if changes were made](#). You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

No additional restrictions — You may not apply legal terms or [technological measures](#) that legally restrict others from doing anything the license permits.

Figure 14: Dataset usage licence - part 2