



Predictive system for characterizing low performance of Undergraduate students using machine learning techniques

E A Ekubo

 **orcid.org/0000-0001-9348-5630**

Thesis accepted in fulfilment of the requirements for the degree
Doctor of Philosophy in Computer and Information Sciences with
Computer Science and Information Systems
at the North-West University

Promoter: Prof B Esiefarienrhe

Co-promoter: Prof N Gasela

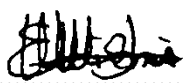
Graduation: July 2020

Student number: 29523389

DECLARATION

I, the undersigned, declare that this thesis, submitted to the North-West University, Mafikeng Campus for the degree of Doctor of Philosophy in Computer Science in the Faculty of Natural and Agricultural Sciences, is my original work except for the citations and I attest that this work has not been submitted to this or any other University for the award of a degree.

Name: **EBIEMI ALLEN EKUBO**

Signature: 

Date: **12-06-2020**

DEDICATION

I dedicate this work to my father, the late Dr Allen Tobin Ekubo, for his love, support and sacrifice.

I love you always Papa.

ACKNOWLEDGEMENTS

I appreciate the guidance, support and encouragement of my supervisor, Professor B.M. Esiefarienrhe, throughout the period of this study. To my co-supervisor, Professor N. Gasela, I appreciate all your input and encouragement.

Dr Nimibofa Ayawei sponsored my studies and I appreciate all you sacrificed to ensure I pursue this degree.

To my mother Luckiere, I appreciate all your prayers and love. To my siblings Tari, Timi, Womotimi, Iyenimi and Ayibaifie, thank you for your love and support.

To my son Ikechukwu, thank you for praying for me and forgiving me whenever I came back from South Africa and had to go back repeatedly, I love you.

To my fiancé Charles, thank you for all your understanding, love, support and prayers.

To my nephews and nieces Favour, Flourish, Fortune, Honour, Douye, Fortress and Allen, thank you for keeping me in your hearts and the cherished moments.

I appreciate Helen Wankasi, Roland, David and Ebipatei, thank you for creating a family away from home in Mafikeng. I cherish all our moments together.

Special thanks to every member of TRC, it was a privilege to share God's presence with you and to all my friends in South Africa and Nigeria for their assistance and support.

Most importantly, to Almighty God, Jesus Christ and the Holy Spirit, you made it possible, may all the glory be unto your name forever and ever. Amen

ABSTRACT

One challenge of educational institutions is the low academic performance of students. This challenge affects students, tutors, institutions and the society in varieties of ways. To deal with this problem, researchers have applied several methods and most recently, researchers have employed data mining methods. This thesis considered the factors that affect low academic performance in Nigeria, employs machine-learning techniques to design models to assist with classification of students' performance and develops a software that classifies students' into different performance groups without the use of data mining tools. The data used for this research was collected from undergraduate students' records from the Niger Delta University, Bayelsa State, Nigeria. The CRISP-DM research methodology was used for the data mining aspect while agile methodology was used for the software development. The modelling was carried out using WEKA tool. Five (5) machine-learning algorithms namely J48 decision tree, logistic regression, multilayer perceptron, naïve Bayes and sequential minimal optimization were used in the data mining to select the algorithm that produces the best model for the data. To analyse the model built by each machine-learning algorithm, six (6) metrics of evaluation namely values of recall or sensitivity, specificity, ROC area, F-Measure Kappa statistics and root mean squared error (RMSE) were used. At the end of the modelling process, the research found the multilayer perceptron as the best classifier for the dataset. This study also considers the use of four feature selection techniques, which are Correlation, Gain Ratio, Information Gain and ReliefF to select the most relevant features out of the 24 features gathered in the dataset. Results from the feature selection procedure selected sixteen (16) most relevant features. Having identified the best classifier for the dataset, the study went further to develop a novel predictive software using php and python programming languages for the implementation of the multilayer perceptron model with the best features identified from the modelling phase. The software is a contribution from this research to enable institutions quickly identify students' performance without prior knowledge of using machine-learning tools. To evaluate the performance of the software, the research used the test dataset and inputted attribute values for each student record. The result from the evaluation process shows the software achieves 98% accuracy, which depicts a high level of dependability.

TABLE OF CONTENTS

DECLARATION	I
DEDICATION	II
ACKNOWLEDGEMENTS	III
ABSTRACT	IV
LIST OF TABLES	X
LIST OF FIGURES	XII
LIST OF ALGORITHMS	XV
ABBREVIATIONS	XVI
LIST OF OUTPUTS	XVII
CHAPTER ONE: INTRODUCTION	1
1.1 Background.....	1
1.2 Nigerian Tertiary Education System	3
1.3 Motivation for this study	4
1.4 Problem Statement.....	5
1.5 Research Questions.....	6
1.6 Research Aim and Objective	6
1.7 Research Design Method.....	7
1.8 Research contributions	9
1.9 Research deliverables	9
1.10 Thesis Structure	10
CHAPTER TWO: LITERATURE REVIEW	12

2.1	Introduction	12
2.2	Educational Data Mining.....	12
2.3	Application of EDM	12
2.3.1	Methods used in EDM.....	13
2.3.1.1	Prediction.....	13
2.3.1.2	Relationship Mining	14
2.3.1.3	Structure Discovery	15
2.3.1.4	Discovery with Models.....	16
2.3.2	EDM Users/Stakeholders and their Benefits	16
2.3.3	The EDM cycle.....	17
2.3.4	Current Challenges of EDM	18
2.3.5	Present and Future of EDM	19
2.4	Data Mining for Predicting Performance	21
2.4.1	Prediction of Employee Performance.....	22
2.4.2	Prediction of Software Performance.....	23
2.4.3	Prediction of Instructor Performance	24
2.5	Data Mining for Academic Performance.....	25
2.5.1	School Dropout and Poor Academic Performance.....	27
2.6	Causes of Poor Academic Performance in Developing Countries	27
2.6.1	A Focus in Nigeria.....	28
2.7	Academic Performance Prediction Modelling	30

2.8	Chapter Summary and Lessons learnt	31
CHAPTER THREE: RESEARCH METHODOLOGY		32
3.1	Introduction	32
3.2	Educational Data Mining Process.....	32
3.3	Framework.....	33
3.3.1	Domain understanding: poor academic performance	35
3.3.2	Data Understanding	35
3.3.2.1	Data Collection	36
3.3.3	Data Preparation Process	37
3.3.3.1	Attribute Selection	38
3.3.4	Modelling.....	40
3.3.4.1	J48 Decision Trees.....	40
3.3.4.2	Logistic Regression	42
3.3.4.3	Multilayer Perceptron (MLP)	43
3.3.4.4	Naïve Bayes Bayesian Classifiers	44
3.3.4.5	Sequential Minimal Optimization (SMO)	44
3.3.4.6	Feature Selection Techniques	45
3.3.5	Predictive System Methodology.....	47
3.3.5.1	Rapid prototyping	47
3.3.6	Evaluation.....	48
3.3.6.1	Metrics of Evaluation in WEKA	49

3.4	Chapter Summary	53
CHAPTER FOUR: DATA MODELLING, RESULTS AND DISCUSSIONS		55
4.1	Introduction	55
4.2	Presentation and Discussions of Results	55
4.2.1	Presentation and Interpretation of Training Dataset.....	56
4.2.2	Presentation and Interpretation of Test Dataset.....	63
4.2.3	Performance of Classifiers and Findings.....	70
4.3	Presentation of Feature Selection	75
4.3.1	Performance Evaluation for Selected Features.....	80
4.3.1.1	Summary of Results.....	85
4.3.2	Performance of Multilayer Perceptron Classifier using the Best Selected Features.....	86
4.4	Chapter Summary	88
CHAPTER FIVE: DESIGN, IMPLEMENTATION AND EVALUATION OF PREDICTIVE SYSTEM		90
5.1	Introduction	90
5.2	The Study Perspective	90
5.2.1	Components of the Predictive System.....	91
5.2.2	The Design Process	91
5.2.3	The System Requirements	92
5.2.4	Sample Model.....	92
5.2.4.1	Sample model design.....	92

5.3	Prototype of the Predictive System	98
5.3.1	Description of the predictive software design	98
5.3.2	Prototype model design	101
5.4	Evaluation of the Predictive System.....	106
5.4.1	Software Evaluation	106
5.4.2	System requirements evaluation.....	109
5.5	Chapter Summary	110
CHAPTER SIX: SUMMARY AND CONCLUSIONS		111
6.1	Introduction	111
6.2	Evaluation of Research Findings.....	111
6.2.1	Research Question One	111
6.2.2	Research Question Two.....	112
6.2.3	Research Question Three.....	112
6.2.4	Research Question Four.....	113
6.2.5	Research Question Five	113
6.3	Summary of conclusions	114
6.4	Challenges and Limitations of this Study.....	115
6.4.1	Data Collection Challenges	115
6.5	Further Research.....	116
REFERENCES		117
APPENDIX.....		137

LIST OF TABLES

Table 3.1: Description of data fields and their respective values.....	39
Table 4.1: The summary of training dataset results obtained from the J48 classifier model.....	57
Table 4.2: The summary of training dataset results obtained from the logistic regression classifier model	58
Table 4.3: Summary of training dataset results obtained from the multilayer perceptron classifier model	60
Table 4.4: The summary of training dataset results obtained from the Naïve Bayes classifier model	61
Table 4.5: The summary of training dataset results obtained from the sequential minimal optimization classifier model	62
Table 4.6: The summary of test dataset results obtained from the J48 classifier model.....	64
Table 4.7: Summary of test dataset results obtained from the Logistic Regression classifier model.....	65
Table 4.8: The summary of test dataset results obtained from the Multilayer Perceptron classifier model	66
Table 4.9: The summary of test dataset results obtained from the Naïve Bayes classifier model.....	68
Table 4.10: Summary of test dataset results obtained from the Sequential Minimal Optimization classifier model	69
Table 4.11: Comparison of the classifier models performance based on correctly and incorrectly classified student data for the training dataset	70
Table 4.12: Comparison of the classifier models performance based on correctly and incorrectly classified student data for the test dataset	71

Table 4.13: Comparison of the classifiers performance on the training dataset using the six selected metrics	72
Table 4.14: Comparison of the classifiers performance on the test dataset using the six selected metrics	74
Table 4.15: Performance of the five classifiers on Correlation ranked attributes.....	81
Table 4.16: Performance of the five classifiers on Gain Ratio ranked attributes.....	82
Table 4.17: Performance of the five classifiers on Information Gain ranked attributes	83
Table 4.18: Performance of the five classifiers on ReliefF ranked attributes	84
Table 4.19: Performance summary of feature selection algorithms used for selecting the best features.....	85
Table 4.20: Summary of multilayer perceptron performance results using the best features dataset with the training dataset	86
Table 4.21: Summary of multilayer perceptron performance results using the best features dataset with the test dataset	87
Table 5.22: Confusion matrix to discern the accuracy of the predictive application on the test dataset	107

LIST OF FIGURES

Fig 1.1: The Research Design Process	8
Fig 2:1: The Educational Data Mining Cycle (Romero et al, 2010)	18
Fig 3.1: The CRISP-DM Process (Olson & Delen, 2008)	32
Fig 3.2: The framework of this research	34
Fig 3.3: The data collection process	36
Fig 3.4: A simple decision tree (Larose & Larose, 2014)	41
Fig 3.5: Structure of a multilayer perceptron with two hidden layers (Kantardzic, 2011).	43
Fig 3.6: The support vector machine showing (a) the separation of the HL class and the LL class with a hyperplane and (b) the point with the highest margin.	45
Fig 3.7: General feature selection process (Dash & Liu, 1997)	46
Fig 3.8: Rapid Application Development model (Kumar & Bhatia, 2014)	48
Fig 3.9: A Confusion matrix	49
Fig 4.1: The J48 classifier model showing the performance of the training dataset	58
Fig 4.2: The logistic regression classifier model showing the performance of the training dataset	59
Fig 4.3: The multilayer perceptron classifier model showing the performance of the training dataset	60
Fig 4.4: The Naïve Bayes classifier model showing the performance of the training dataset	62
Fig 4.5: The sequential minimal optimization classifier model showing the performance of the training dataset	63
Fig 4.6: The J48 classifier model showing the performance of the test dataset	64

Fig 4.7: The Logistic Regression classifier model showing the performance of the test dataset.....	66
Fig 4.8: The Multilayer Perceptron classifier model showing the performance of the test dataset.....	67
Fig 4.9: The Naïve Bayes classifier model showing the performance of the test dataset.....	68
Fig 4.10: The Sequential Minimal Optimization classifier model showing the performance of the test dataset.....	69
Fig 4.11: Summary of the classifiers performance on the training dataset using the six selected metrics	73
Fig 4.12: Summary of the classifiers performance on the test dataset using the six selected metrics	75
Fig 4.13: Correlation ranked features from the most important to the least important.....	77
Fig 4.14: Gain Ratio ranked features from the most important to the least important.....	78
Fig 4.15: Information Gain ranked features from the most important to the least important	79
Fig 4.16: ReliefF ranked features from the most important to the least important	80
Fig 4.17: The multilayer perceptron model built with the best features using the training dataset.....	87
Fig 4.18: The multilayer perceptron model obtained with the best features using the test dataset.....	88
Fig 5.1: Use case diagram showing the Faculty Officer’s roles in using the predictive system.....	93
Fig 5.2: Context diagram showing the data process and flow within the system	93
Fig 5.3: Welcome screen.....	94
Fig 5.4: Login Screen	95

Fig 5.5: Sample design of predictive application for student information form	96
Fig 5.6: Sample design of predictive application for result prediction	97
Fig 5.7: Welcome Screen	102
Fig 5.8: System Login Page	103
Fig 5.9: Prediction Application showing the user input.....	104
Fig 5.10: Prediction Application showing the prediction results	105
Fig 5.12: Cross-section of the test dataset showing student records, actual result and predicted result obtained from the Prediction Application	106

LIST OF ALGORITHMS

Algorithm 4.1: Resampling of dataset.....	55
Algorithm 5.1: The design process	91

ABBREVIATIONS

CGPA	–	Cumulative Grade Point Average
CRISP-DM	–	Cross-Industry Standard Process for Data Mining
EDM	–	Educational Data Mining
ML	–	Machine Learning
MLP	–	Multilayer Perceptron
NDU	–	Niger Delta University
WEKA	–	Waikato Environment for Knowledge Analysis

LIST OF OUTPUTS

Ebiemi Allen Ekubo and Michael Bukohwo Esiefarienrhe. Attributes of low performing students in e-learning system using clustering technique. IEEE Xplore. Volume 1, Pages: 1324-1328, 2019. DOI:10.1109/CSCI49370.2019.00247

Ebiemi Allen Ekubo and Michael Bukohwo Esiefarienrhe. Predictive system for characterizing low performing undergraduate students using machine-learning techniques. Australasian Journal of Information Systems. *Article under review.*

CHAPTER ONE: INTRODUCTION

1.1 Background

The consequences of low academic performance by undergraduate students can be long-term which is often exhibited as anxiety (Nurmi et al, 2003), low self-esteem (Aryana, 2010), and fear of failure (Nsiah, 2017). Students with low academic grades often feel frustrated and resort to dropping out of learning institutions (Stinebrickner & Stinebrickner, 2014) or struggle and risk staying in school for extended period periods (Shannon & Bylsma, 2006). Poor academic performance also has its effects on educational institutions and the society; for institutions, poor academic performance of students curtails the proper execution of educational operations and it reduces the amount of available manpower in different fields (Al-Zoubi & Younes, 2015). This challenge of poor academic performance is found in almost every part of the world; however, in a developing country such as Nigeria, many universities record a high number of low performing undergraduate students (Oyebade & Dike, 2013) which are attributed to factors such as poor secondary school background, lack of students' commitment and environmental factors (Bolapeju et al, 2014). The studying conditions in Nigeria are so poor that many students that begin a course drop out before graduating and a high number of students that complete their studies graduate with weak quality degrees.

In dealing with the challenge of poor academic performance, researchers have studied factors associated with low performance in different countries and at different educational levels (Mushtaq & Khan, 2012). These researches have designed models using data mining techniques to assist students perform better, improve methods of teaching and generally provide educational institutions with better methods to aid students engage in learning and improve learning outcomes (Ocumpaugh et al, 2014). However, these models have been designed and implemented in only a few learning environments, which are largely in developed countries (Guri-Rosenblit, 2006). For a developing country such as Nigeria, the educational data mining research done has focused on predicting student performance using available attributes. For example, Adeyemo & Kuye (2006) predict student performance using attributes such as students' demographics and previous academic scores while Oyerinde & Chia (2017) combine scores from different courses to predict student academic performance. However, there is no empirical record of improvement of students' performance or model developed to aid in improving students' academic performance. Ololube (2013) states that a major challenge with developing models to improve learning outcomes in the

country is that many Nigerian universities lack the technological systems used in modern educational settings. Undeniably, most Nigerian universities do not have systems to monitor students' learning behaviours or discern students' engagement levels in class. Nevertheless, these universities could start by developing and implementing a system that classifies students based on their academic performance and identify new students at risk of poor performance using the available features. These institutions could then use this information in making decisions and creating intervention measures to improve the academic performance of their students. To achieve this, these institutions must understand the factors that influence the low performance of their students by collating student attributes, modelling these into a system, providing intervention support systems and creating an enabling environment for these methods to thrive.

In view of this, this study approaches the phenomenon of low academic performance by looking at the attributes of low performing students in Niger Delta University (NDU), situated in Bayelsa State, Nigeria. Prior research indicates that this university has no information management system or model in place to identify low performers or to improve student performance. Hence, this research serves as a foundation where future researchers could build upon in creating a more robust system. Furthermore, to achieve the purpose of the study, the study considers only students with cumulative grade point average (CGPA) of less than 3.0 and categorises them into two distinct groups, which are low risk students with CGPA between 2.50-2.99 and high-risk students with CGPA below 2.50. This study makes use of the 3.0 benchmark because it assumes that students above 3.00 are students who perform well and are able to pursue a postgraduate degree after initial graduation. However, students with low CGPA often find it difficult to pursue a postgraduate degree as an average Nigerian University requires a student to have a CGPA of at least 3.00 to qualify for admission. For the two groups used in this study, students categorised as "low risk" are students with good grades, yet require some form of intervention to help them perform better. The "high-risk" group are students that are more likely to drop out from the university or stay on longer due to their poor grades; these set of students require major intervention for them to continue with their education and improve on their grades. Hence, this study strives to build a predictive system which could assist the Niger Delta University identify students at the risk of failure. This system, when fully developed, can identify a new student as either low risk or high risk and with that information, the university could generate and develop support systems to assist early enough.

The next session examines the general Nigerian tertiary education system and the way it functions.

1.2 Nigerian Tertiary Education System

The Nigerian tertiary education system comprises universities, polytechnics and colleges (WES Staff, 2017). Over 150 universities are currently operational in the country, owned by either federal government, state government or private individuals (NUC, <http://nuc.edu.ng/nigerian-universities/>).

To gain admission into any Nigerian tertiary institution, the applicant must meet the following requirements (WES Staff, 2017)

1. Obtain a minimum of five credits including Mathematics and English from their senior secondary certificate examination.
2. Obtain a minimum cut-off score from the Unified Tertiary Matriculation Examination (UTME) organized by the Joint Admissions and Matriculation Board (JAMB)
3. Obtain a minimum post-UTME cut-off score for the course of study in the institution where the student is seeking admission

The National University Commission of Nigeria (an organisation in charge of overseeing the administration of higher degree education in the country) offers a five-point grading system, which is the grading and degree classification system that Nigerian universities are required to use (WES Staff, 2017). Below is a brief description:

- | | | |
|----------------|---|-----------------------------|
| a. 4.50 – 5.00 | – | First Class |
| b. 3.50 – 4.49 | – | Second class upper division |
| c. 2.40 – 3.49 | – | Second class lower division |
| d. 1.50 – 2.39 | – | Third class |
| e. 0.00 – 1.49 | – | Fail |

This grading system followed by Nigerian universities shows that students with CGPA of 3.00 and below are classified in the lower divisions and graduates who obtain that class often find it difficult to secure admission for postgraduate degrees in Nigeria. In many instances, they often have to study for postgraduate diploma courses before they can further their education. With the high

unemployment rate in the country, most graduates tend to pursue postgraduate degrees to increase their chances of gaining employment but with their poor academic degree, it is often a difficult feat to achieve.

Thus, this study uses data collected from the Niger delta university, which is a state owned university in Nigeria with over 10,000 students (NDU, <http://www.ndu.edu.ng/nduprofile.html#>) and focuses on students with $GPA < 3.00$.

1.3 Motivation for this study

The educational data mining community has developed systems that monitor and interpret student learning behaviours with applications in improving student models, discovering domain models, studying support offered by learning software and scientific discovery of learning and learners (Baker, 2010). These systems have shown improvements in student learning outcomes and assisted stakeholders in making informed decisions (AlShammari et al, 2013). These systems, however, are yet to spread across different learning environments and institutions (Romero & Ventura, 2013). This is due to challenges such as lack of adequate knowledge by instructors and managers, ethical issues, government policies, low funding and ineffectual management of the systems (Meenakumari & Kudari, 2015; Liñán & Pérez, 2015). Yet, poor academic performance is a major concern for educational institutions, and stakeholders continually seek ways to curb the problem (Katamei & Omwono, 2015).

In Nigerian universities, the rate of poor academic performance is on the increase, which could be attributed to several factors unique to the Nigerian society. Phlegmatic performance invariably leads to high dropout rates, which in turn increases the rate of crime in the country (Ajaja, 2012). In addition, the policies designed to improve student performance are not working in the country (Babalola, 2015) and Nigerian institutions need to tap into the development of models using machine-learning techniques to intervene and improve students' performance.

The motivation to carry out this research originates from three distinct problems:

1. A palpable increase in poor academic performance in Nigerian universities.
2. Ineffective measures to curb poor academic performance are a significant challenge for tertiary education.
3. There are no educational models in place to assist low academic students in Nigeria.

The factors that influence low academic performance established across different developing countries and factors distinct to Nigeria and Nigerian students are relevant in this study to highlight the causes and effects of low performing students in the country. Concisely, this research strives to create the opportunity for future researchers to develop methods and models that monitor student learning behaviours and learning outcomes in Nigeria.

1.4 Problem Statement

Low academic performance is a challenge for every institution in society and this severely affects the goals of these educational institutions, which is to prepare their scholars for the society by providing quality education that ultimately allows them compete favourably in the society (Berkowitz, et al, 2017). This low academic performance challenge also affects institutions as universities that record high rate of poor academic performance receive low university rankings on global scales (Olcay & Bulu, 2017; Vernon et al, 2018). Furthermore, tertiary institutions regularly come up with policies to enhance their growth, thus they are constantly looking for effective and efficient methods that could create improved policies for their institutions. As stated earlier, low academic performance cuts across every society; however, the challenge is more prominent in developing countries, which has low-income earners, poor access to good medical care, poor electricity and poor funding that only complicate the performance capacities in their intakes (Muralidharan, 2017; Kim et al, 2019).

Research in recent times has used data mining techniques to gain knowledge about students and their learning patterns, yet scholars have not successfully designed robust and informed models for developing countries (Vahdat et al, 2015; Kassarnig et al, 2018). Although some good models exist for scholars in developed countries, it is necessary to design models for developing nations, as the attributes of low performance often vary with the specific contextual factors in every society. Using data mining methods, organizations gain previously unknown knowledge from huge sets of data (Milovic & Milovic, 2012) and since educational institutions regularly produce huge amount of data, this fits quite well. Hence, this research interrogates the possibilities and practicalities of employing machine-learning methods to classify students with low academic performance in a Nigeria as a developing country.

To achieve this goal, this research follows the method of identifying key attributes of low academic performance in Nigeria, comparing the performance of five different machine-learning algorithms, selecting the best features from the entire attributes collected, selecting the best classifier model

and developing a predictive software using the best classifier model identified. This proposed software provides the university with timely and accurate information to identify low performers and assist the university intervene early enough. This research utilises data collected from the Niger Delta University, a public university in Bayelsa state, Nigeria, to achieve the objectives of the research. The development of the predictive system is the most novel contribution of this thesis to the body of knowledge and serves as a platform to solve the problem associated with identifying learners that perform poorly in higher education for developing countries.

1.5 Research Questions

The specific research question is “How could the use of machine learning techniques aid in modelling a predictive system for the classification of low performing undergraduate students in NDU?”

Specific subsidiary research questions considered by this research are as follows:

1. Which factors are associated with low performance of undergraduates in Nigeria?
2. How could these factors be collected and represented in machine-readable format for data mining?
3. Which machine learning technique could best classify low performing students?
4. What are the best sets of features from the total features collected for predicting and intervening in low academic performance?
5. Can the best machine learning technique and best features identified assist in the design of a predictive system to identify low performing students?

1.6 Research Aim and Objective

Aim

This research aims to identify and classify the causes, effects and probable solutions to underperformance of undergraduates in Nigerian higher educational institutions.

Objectives

The objectives of this research are to:

1. Examine and describe factors affecting underperformance of undergraduates in Nigeria by reviewing literature extensively;
2. Collect low performing students' data in NDU based on factors identified from literature using data capturing techniques and convert the data from source documents to machine readable format using Microsoft Excel;
3. Identify the best machine learning technique for classifying low performers in NDU by analysing five machine learning algorithms for classification, which are J48, LR, MLP, NV and SMO;
4. Select the best features from the dataset using four feature selection techniques, which are Correlation, Gain Ratio, Information Gain and ReliefF; and
5. Utilise the best machine learning algorithm and the best features identified to design a predictive system for identifying low performers in NDU using PHP programming language.

1.7 Research Design Method

The research design for this study used the Cross-Industry Standard Process for Data Mining (CRISP-DM) and the diagram in Fig 1.1 illustrates the complete design process. From the diagram, the six CRISP-DM steps followed in this study are domain understanding, data understanding, data preparation, modelling, evaluation and deployment (Chapman et al, 2000). In line with the CRISP-DM process, the first step is gaining a background understanding of the factors that influence low performance of Nigerian undergraduate students through survey of literature. This information privileged the gathering of data into an Excel worksheet to gain a good understanding of the data. Next, the data preparation stage involved cleaning and preparing the data for modelling and the modelling process employed the WEKA modelling tool, which has several classification algorithms for producing different models from the data. The evaluation stage of the models produced assisted in determining the model with the best set of features that generalises the data. Finally, the deployment phase involved the design and implementation of a predictive system to identify students with low performing attributes using the best model identified. This deployment stage looks at gathering the requirements for the design of the predictive system, implementing the best model and features identified from the evaluation stage and evaluating the system designed to ensure that it fulfils the aim of the study.

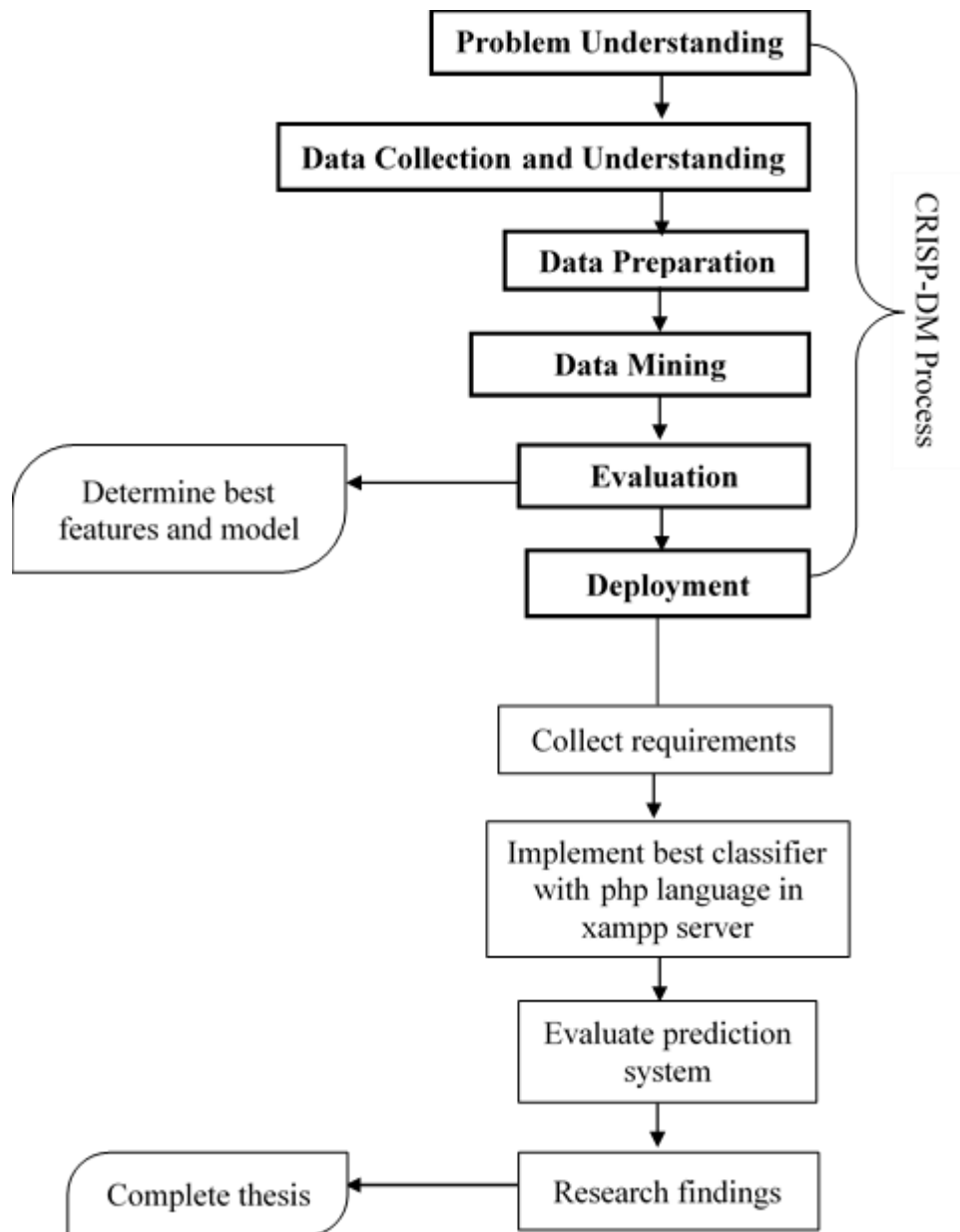


Fig 1.1: The Research Design Process

The findings from the entire research design process contribute to the new knowledge generated in the thesis. Furthermore, the predictive system designed used data collected from the university that was the site in this study and it is specifically designed based on the features collected on the students. Other institutions could use this framework to design predictive models based on their unique intake and student attribute, considering of course their specific needs.

1.8 Research contributions

One challenge identified in EDM is the lack of generalised models, especially as research carried out shows that models and advancements concentrate more on the western countries, yet the developing countries are not part of the research findings; hence the models lack applicability and context (Baker & Yacef, 2009; Baker, 2010; Vahdat et al, 2015). Therefore, this research aim to contribute to the body of knowledge in the EDM community of practice by specifically focusing on developing models contextualised and designed for developing countries.

This research also contributes to new knowledge in the following innovations:

1. The university site of this study has no software in use for monitoring students' performance; therefore, the designed software would serve as a novel design that provides a foundation for researchers to analyse students' performance. This novelty and initiative could open up other opportunities for future research.
2. The study provides a prototype model that identifies students at risk of failing; this model is modifiable for use in other learning institutions and should be robust enough to assist educational stakeholders in reducing the failure/dropout rates.
3. The identification of the most efficient machine-learning algorithm for identifying and classifying low performing students in tertiary institution databases. The identified algorithm is selected after comparing five-(5) classification algorithms on various indices of performance.
4. The development of a software to implement the identified algorithm that is installed directly by institutions without the use of any data-mining package. This is vital as the use of data mining packages introduces unnecessary steps that are time consuming and thus costly in terms of resources.
5. This thesis develops the interface for data capturing of individual student records for the process for the selected machine-learning algorithm. This enables each student performance to be assessed and reported.

1.9 Research deliverables

The research deliverables are as follows:

1. This thesis develops a novel machine learning software to implement the multilayer perceptron algorithm with customised data capturing capabilities for individual students

2. The design and implementation of the software shall be systematically developed for published academic papers
3. The specification of the problem, literature review, research methods and development of the software shall constitute a final PhD thesis submitted for the same qualification.

1.10 Thesis Structure

The thesis follows the structure outlined below:

Chapter 2: Literature Review

This chapter describes diverse perspectives on educational data mining and reviews literature in the following areas: data mining for predicting performance, data mining for academic performance, school dropout and poor academic performance, causes of poor academic performance in developing countries with a focus in Nigeria, and academic performance prediction modelling. The review identifies gaps and challenges in previous and related studies, indicating specifically the niche that this study fills.

Chapter 3: Research Methodology

This chapter presents the methodology followed in undertaking this research, which is the Cross-Industry Standard Process for Data Mining (CRISP-DM) and the framework followed to accomplish the objectives of this research. Some specific areas examined in this chapter include the process of collecting and collating data and the preparation of data for mining, discussion of the five-machine learning algorithms selected for the modelling process, techniques for feature selection and techniques for evaluation of the models.

Chapter 4: Data Modelling, Results and Discussions

This chapter presents the results from the data modelling process using the WEKA software. It investigates modelling the dataset collected for the research by applying five machine learning algorithms namely, J48, logistic regression, multilayer perceptron, naïve Bayes and sequential minimal optimization to select the best classifier model for the study. This chapter also presents the results of using four feature selection algorithms called Correlation, Gain Ratio, Information Gain and ReliefF to select the best features within the dataset.

Chapter 5: Design, Implementation and Evaluation of the Predictive System

This chapter presents the design, implementation and evaluation of the predictive system, which serves as the final (deployment) stage of the research methodology (CRISP-DM) in this study. It presents the specifications and requirements of the system, the design of a sample model for the predictive system, the prototype design of the predictive application and finally evaluation of the designed software.

Chapter 6: Summary and Conclusions

This chapter provides a summary of the entire research and succinctly evaluates the contribution of the research to the body of knowledge in IT, discussing the challenges and limitations of the study, and offering recommendations for future research.

CHAPTER TWO: LITERATURE REVIEW

2.1 Introduction

This review explores the relevant and most recent literature on educational data mining, its application, methods, benefits, challenges and future prospects. The chapter specifically interrogates how data-mining techniques assist in predicting performance in different areas and holistically predicts the performance of students. The review concludes with a focused discussion on the causes of poor academic performance of undergraduate students in developing countries with a focus in Nigeria.

2.2 Educational Data Mining

The application of data mining techniques in education is a developing multidisciplinary research area termed Educational Data Mining (Romero & Ventura, 2013). Educational Data Mining (EDM) as a research area critically focuses on developing methods from the unique data available in educational settings (Romero et al., 2010). Educational data is found in different sources within diverse learning environments, which regularly produce large amounts of data (Romero & Ventura, 2013). EDM strives to gain knowledge from large datasets (Han et al, 2011) and with the vast and unique educational data available, employing data mining techniques to understand learners and improve learning process (Algarni, 2016). Since education is a stimulus for the growth of any society and a society thrives socially and economically when its education system is on the right track (Mitra, 2011), thus employing EDM techniques benefits the society and essentially improves learning, which is measured through improved performance of learners and the learning processes (Romero & Ventura, 2013).

2.3 Application of EDM

The major areas of applications in EDM outlined by Romero et al (2010) are:

- *Communicating to stakeholders:* The goal here is to use the knowledge gained from EDM process to assist stakeholders in evaluating the activities of students and their course practices.

- *Maintaining and improving courses*: The aim is to assist educators identify ways of improving course content and activities from the knowledge gleaned from students' learning habits.
- *Generating recommendation*: The interest is to recommend relevant content to students working on a particular course to assist in their learning and learning outcomes.
- *Predicting student grades and learning outcomes*: The focus is to use data from students learning activities to predict student grades or learning outcomes. This research focuses on this application since the goal is to determine students' learning outcomes from available educational data in NDU.
- *Student modelling*: The goal is to build a student model from the knowledge gathered from students' learning habits; usually encompassing features such as learning styles, motivation, preferences, learning progress and emotional states of students.
- *Domain structure analysis*: The aim is to discover the value of a domain structure model by measuring its ability to predict student performance.

Other applications of EDM identified by Baker (2010) entail studying the instructional support offered by educational software and generating scientific innovations about learners and learning.

2.3.1 Methods used in EDM

Baker & Inventado (2014) identified the popular methods used in mining educational data as prediction, relationship mining, structure discovery and discovery with models. These methods, according to them, show more promise and most researchers in the EDM domain have succeeded in deploying these methods. The following segment describes these methods in some detail.

2.3.1.1 Prediction

Prediction methods aim to develop a model that deduces a single part of the data (predicted variable) from combinations of other parts of the data (predictor variables) following the directions offered in Sachin & Vijay (2012); and Aziz et al (2013). These models assist in predicting a value in situations where it is not necessary to find a label for the concept. It also helps identify concepts connected to the prediction of another notion. Common prediction methods in EDM are classification, regression and latent knowledge estimation.

1. **Classification:** In classification, the value of the predicted variable can be either binary or categorical. In EDM, classifiers are normally authenticated using cross-validation by reserving a portion of the dataset for evaluating the accuracy of the model. Popular classification methods used in EDM are decision trees, decision rules, random forests, step regression, multilayer perceptron and logistic regression.
2. **Regression:** In regression, the value of the predicted variable is a continuous variable. Linear regression and regression trees are the popular regression models used within the EDM domain. The model produced using this method in EDM is the same as in statistics; however, the process of selecting and validating the model in EDM is different.
3. **Latent Knowledge Estimation:** In latent knowledge estimation, the purpose is to measure students' knowledge of skills and concepts by evaluating their accuracy levels. Through these methods, measuring knowledge directly is not possible but inferred from students' performance. This process of deducing students' knowledge assists in providing solutions to some pertinent EDM questions. The models used for latent knowledge estimation come from either new idea in classical psychometric approaches or user modelling/artificial intelligence research and the algorithms used for latent knowledge estimation are Bayes Nets, Bayesian Knowledge Tracer, logistic regression and performance factors assessment. However, for large datasets, combining multiple approaches can be more effective than using a single method.

2.3.1.2 Relationship Mining

Relationship mining determines connections between variables in a dataset that contains a range of variables. This might take the form of finding out the strongest associations of variables with a particular variable or discerning which associations between two variables are the strongest. The four types of association mostly used in EDM are association rule mining, sequential pattern mining, correlation mining and causal data mining.

1. **Association Rule Mining:** Association rule mining discovers 'if-then' rules, which usually predicts a specific value based on the combination of a set of values. This method reveals general existence in data, which would have been manually challenging to discover.
2. **Sequential Pattern Mining:** Sequential pattern mining establishes temporal relationships amongst events. The classical sequential pattern mining and motif analysis are two models

used to find sequential patterns. With many possible patterns discovered at the end of the modelling process, some parameters are necessary in selecting the valuable rules for output.

3. **Correlation Mining:** Correlation mining searches for positive or negative linear relationships between variables, which is also a familiar goal in statistics. In EDM, researchers have used correlation mining to determine relationships between student attitudes and behaviours such as gaming the system or requesting assistance (Baker et al, 2008).
4. **Causal Data Mining:** Causal data mining determines if one occurrence resulted in the occurrence of another. Causal data mining finds actual relationships by viewing patterns of covariance amongst variables in the dataset. Causal data mining use in EDM domain assisted researchers to predict factors that could lead students performing poorly (Fancsali, 2012) and to clarify how attitudes and sexual behavioural patterns affect performance and learning outcomes in an intelligent tutor system (Rai & Beck, 2011).

2.3.1.3 Structure Discovery

Structure discovery aims to determine structure from data without ground truth or knowledge of what the finding would be like. This method contrasts prediction models where ground truth is required before model development can occur. The structure discovery field originates from the discipline of psychometrics and educational measurement. Structure discovery algorithms commonly used in EDM include clustering, factor analysis and domain structure discovery.

1. **Clustering:** Clustering finds naturally grouped points within data by dividing the entire dataset into a set of clusters. Clustering is suitable for circumstances where there is no prior knowledge of the groups in the dataset. An ideal set of clusters creates a cluster with a data point similar to data points within its group than the data points in other groups. Examples of clustering algorithms are hierarchical agglomerative clustering (HAC), k-means, Gaussian mixture modelling (EM-based clustering), and spectral clustering.
2. **Factor Analysis:** Factor analysis aims to discover natural clusters of variables (instead of data points) into a group of factors not easily observed. In EDM, factor analysis assists in dimensionality reduction, reducing the possibility of overfitting, and determining meta-

features. Algorithms used in factor analysis include principal component and exponential-family principal component analysis.

3. **Domain Structure Discovery:** Domain structure analysis aims to discover the structure of knowledge within an educational domain such as determining which course content links to particular skills across students (Tam et al, 2015). In EDM, domain structure discovery assists researchers to test data (Desmarais, 2011) and track learning in an intelligent tutoring system (Cen et al, 2006). Algorithms used in domain structure discovery include purely automated algorithms and methods that make use of human judgement in the model discovery process such as learning factor analysis.

2.3.1.4 Discovery with Models

In discovery with models, the logic is to use a model developed through prediction, clustering or knowledge engineering as a part of a second analysis or model as in prediction or relationship mining. In EDM, a common method of applying discovery with models is by making use of the predictions from an initial model as the predictor variables in a different prediction model. Discovery with models often influences the generalization of a prediction model across different situations.

2.3.2 EDM Users/Stakeholders and their Benefits

Romero & Ventura (2013) identified the stakeholders of EDM as learners, educators, researchers and administrators. These users play different roles in the system through their inputs and expected outputs. Below are descriptions of their roles and benefits in the EDM system.

1. **Learners:** Students interact actively with any educational system; they offer data ranging from demographic information, learning pattern, process and outcomes, and interaction with other learners and instructors through traditional means or computer-based methods. Learners can benefit from EDM as this platform provides support for learners to reflect on their learning processes and outcomes, responding to the needs of learners, offering learners standard recommendations and feedback, and generally developing methods to increase the performance of learners.
2. **Educators:** Educators provide instructions for learners, offer course outlines, review learners learning process through quizzes, tests, assignments and examinations, and

understand learners' behaviour through interactions. Educators can benefit from EDM by reflecting upon and improving on their methods of instruction, organizing course curricula, attempting to know their students' learning processes and understanding their social and mental behaviours. With such knowledge acquired from EDM process, educators can identify areas that students struggle with and modify their teaching methods.

3. **Researchers:** Researchers contribute to the advancement of EDM by developing, evaluating and comparing data mining techniques to recommend the most appropriate and suitable for each particular educational task and assessing the learning efficiency. The annual International Conference on Educational Data Mining launched in 2008 and the *Journal of Educational Data Mining* established in 2009 with current EDM interests encourage researchers to focus on relevant topics that promote the EDM community.
4. **Administrators:** Administrators are concerned about the growth of institutions; they are members of faculties and advisors within institutions that are in charge of distributing funds for the smooth operations in institutions. Administrators are the managers that require correct and timely information in making the best decisions for tutors and learners. EDM can offer such personnel knowledge to evaluate the best methods of promoting the institution and distributing human and material resources in the institution.

2.3.3 The EDM cycle

Applying data mining techniques in educational systems is an iterative series of constructing hypotheses, testing and improvement (Romero & Ventura, 2007). The diagram (Fig 2.1) shows the iterations of applying data mining in educational systems (EDM cycle). The knowledge acquired from the mining process should aid in decision making by returning into the cycle of the system for improvement.

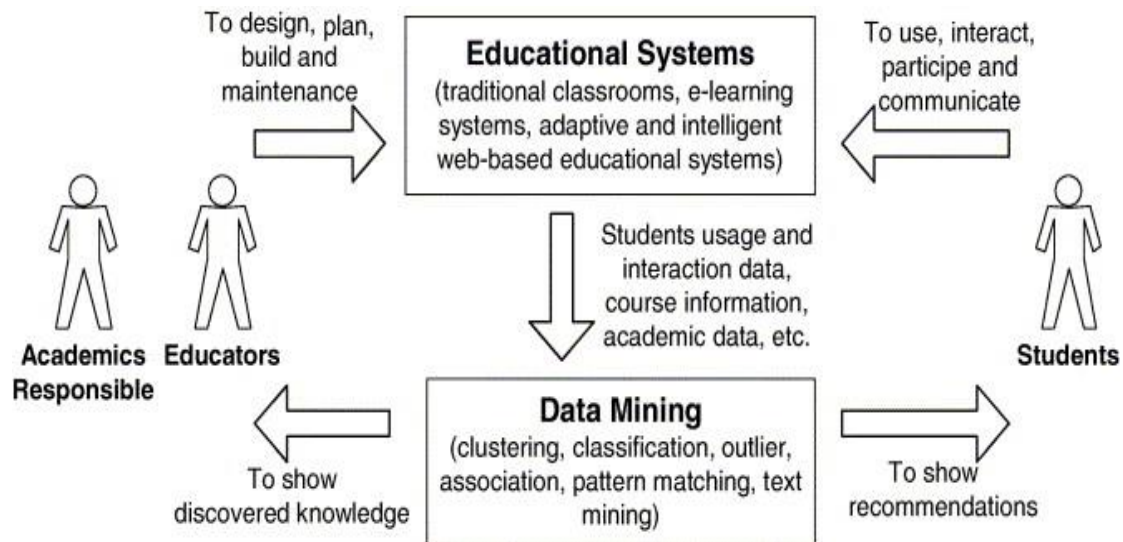


Fig 2:1: The Educational Data Mining Cycle (Romero et al, 2010)

From the diagram, the EDM cycle shows that the educators and academics are responsible for the designing, planning, building and maintaining of educational systems while the students interact with the system. Using data mining techniques like classification, clustering, association mining and with all the existing information about students, courses and interactions within the system, it is possible to discover valuable information that could improve the educational systems and assist students perform better. The knowledge from this process could assist students through enhanced accessibility of recommendation systems. Subsequently, educators could effectively monitor students and evaluate course structure and administrators could equally improve the effectiveness of the educational systems and make it flexible for the users.

2.3.4 Current Challenges of EDM

Acquiring valuable knowledge using data mining techniques in any educational system is likely to improve its current state. However, it is necessary to consider the challenges encountered by EDM users, researchers and the EDM community. Descriptions of challenges observed within the EDM domain are itemised below.

- **Cost:** With the advent of big data, the associated cost of storage and retrieval is a big concern for many organizations especially in developing countries (Luna et al, 2014). Educational institutions planning to implement EDM applications must consider the storage cost and the cost of employing knowledgeable staff to manage the systems (Bienkowski et al, 2012; Vahdat et al, 2015).

- **Generalisation:** With EDM, it is difficult to develop a general method for all educational environments because of the diverse variables in different environments. Research carried out in EDM also shows that models and advances have been more robust in western countries and many developing countries have not been a part of the research or findings (Baker & Yacef, 2009; Baker, 2010; Vahdat et al, 2015).
- **Privacy:** Data privacy of individuals in data mining has been a major concern lately (Smith et al, 2012). In EDM, individual student privacy has also raised concerns specifically with young learners who are unable to protect their privacy by giving necessary consent (Sabourin et al, 2015). With this challenge in mind, developers of EDM tools must consider methods of safeguarding individual privacy of students.

2.3.5 Present and Future of EDM

From the foundation of EDM, the goal of its domain is to provide relevant educational resources for stakeholders in improving the education system (Bienkowski et al, 2012). In some way, this goal has scored significant achievements through breakthroughs celebrated and in other ways; the goal gives birth to more concerns and ideas. The breakthroughs that the EDM community celebrates so far are the developments of some tools and models specifically developed for educational data such as decisional tool (Selmoune & Alimazighi, 2008), LiMS (MacFadyen & Sorenson, 2010), EDM Workbench (Rodrigo et al, 2012), Moodle Data Mining (MDM) Tool (Luna et al, 2017) etc. There is also the development of behavioural patterns like gaming the system (Baker et al, 2004), Off-task behaviour (Baker, 2007), WTF behaviour (Wixon et al, 2012), etc.

Data mining application in education in both traditional and computer-based educational systems seeks to improve the education system; however, the data sources and objectives are different and require the use of different methods to acquire data and gain knowledge from the collected data (Romero & Ventura, 2013). The traditional classroom records data mostly through traditional methods of instructing, recording and monitoring students (Romero & Ventura, 2020). Computer based educational systems, which consist of web based educational systems, learning management systems, intelligent tutoring systems, and adaptive and intelligent hypermedia systems make use of the computer to instruct, evaluate and monitor learners, their learning patterns and their learning outcomes (Romero & Ventura, 2013).

From the survey of Peña-Ayala (2014), six EDM approaches developed over the years are student modelling, student behaviour modelling, student performance modelling, assessment, student support and feedback, and curriculum-domain knowledge-sequencing-teaching support.

- Student modelling: focuses on representing how students adjust to the learning process to meet particular learning requirements. Student modelling seeks to develop ways to improve the education domain of students by looking at features such as learning patterns, accomplishments, emotions, learning preferences and skills.
- Student behaviour modelling: aims to define and predict specific attitudes of learners to align the system to the learning trends. It focuses on modelling behaviours such as requesting assistance, guessing, gaming, examining, willingness to work in a team, etc.
- Student performance modelling: the major concern is to predict how well a student can complete specific learning tasks. Pointers that assist in modelling student performance are accuracy, productivity, time, resource used, proficiency, inadequacies, etc.
- Assessment: centres on distinguishing students' learning abilities by testing their acquired skills through questioning, evaluating their views and reflections.
- Student support and feedback: focuses on offering support to and feedback from learners. Support given to learners is bound to improve their performance or correct their errors. Feedback from students could assist them in evaluating the system and making recommendations.
- Curriculum-domain knowledge-sequencing-teaching support: centres on offering efficient ways for educators to deliver knowledge and provides support for them to effectively monitor students, search content, create collaborations and evaluate their teaching methods and outcomes.

A better understanding of the current causes, effects and improvements of educational systems is part of the expectations that EDM is likely to inaugurate in the future; however, achieving this calls for the support of all educational stakeholders (Sukhija et al, 2015; Berendt et al, 2017). It is important to build an educational environment where there is trust for EDM research to grow effectively (Sukhija et al, 2015). The growth of EDM also depends on the advancement of

computer-based learning and accessibility of data (Bakhshinategh et al, 2018). Some important areas the future of EDM research needs to look into identified by Sukhija et al (2015) are:

- Acquiring large and well-structured datasets: It is important for EDM research to provide ways to acquire detailed and well-structured datasets from any educational environment. The computer-based learning environment provides easy ways to collect large datasets from its environment, but other environments require sophisticated tools and knowledge that takes time and money. An EDM tool that can easily integrate with all learning environments is definitely important in the future of EDM (Vahdat et al, 2015).
- Creating resourceful datasets: Many researchers face the problem of resourceful datasets, which compels them to explore into other methods or make use of datasets that might not be useful for the research. The future of EDM needs to integrate useful datasets to design a flexible system for implementation across all learning environments.
- Merging of methodologies: There is need to combine different algorithms to create a hybrid technique. Most researchers use methods in isolation; combining different effective methods could improve the performance of EDM systems (Siemens & Baker, 2012).
- Credibility of EDM: The future of EDM must be concerned with developing systems that are in line with policies of education systems in different learning environments and creating user-friendly and dependable systems for users.
- Studies on comparative techniques: Research opportunities for the comparative study of different data mining techniques used in EDM is available for future researchers. Comparing and contrasting different techniques could create sustainable approaches for other researchers to deploy relevant technique based on their mining tasks.

2.4 Data Mining for Predicting Performance

Data mining combines different areas such as machine learning, statistics, pattern recognition, artificial intelligence, database technology and visualisation (Kantardzic, 2011; Tan et al 2013; Zaki et al, 2014) to extract meaningful information from huge sets of available data (Han et al, 2011). The information extracted from the data mining process assists organisations in decision-making that improves their business strategy and ultimately increases their business performance (Kasemsap, 2015). Data mining practice records improvements in various fields that has made it

popular and increasingly sought after. One major area of use across all sectors is in predicting the performance of systems or system users; thus, this research delves into the use of data mining in predicting the causes of low performance of students in their course of study and what necessary precautionary steps to be taken with the information available to stakeholders.

2.4.1 Prediction of Employee Performance

Every organization needs a strong network of employees that add value to the organization. Developing human resources is a major concern for executives in every business sector as the process of selecting and managing the right employees are of great interest to them. Immediately after the employment of new staff, managers become concerned about their performance and still have to evaluate these employees for future purposes (Shields et al, 2015).

With the huge amount of data available in every organization due to the use of automatic systems for almost every task and in almost every area in organizations, these organizations seek ways to make accurate and timely decisions (Henke et al, 2016). The use of data mining techniques assists in evaluating and summarising important knowledge of diverse views from data gathered (Henke et al 2016; Kirimi & Moturi, 2016).

Using data mining techniques in managing human resource is an emerging domain; from the review of Strohmeier & Piazza (2013), the human research management domain still requires specific methods to enhance the evaluation of performance in line with legal principles. Some notable research carried out within this domain are talent forecasting, employee performance prediction, predicting training needs of employees, and talent management support.

The major data mining technique used in predicting performance in this domain is the classification method. Kirimi & Moturi (2016) used the decision tree algorithm to predict employees' performance based on their previous assessment records. Jantan et al (2009) developed an architectural framework to forecast employees' talents from experience data. This framework could assist organisations select the right talent for the right task. Valle et al (2012) used the naïve Bayes classifier to predict the performance of sales agents in a call centre and they concluded that operational features play a major role on their future performance than their individual or socio-economic attributes. They conclude that employers must select sales agents based on their performance. Al-Radaideh & Al Nagi (2012) used real data gathered from different companies and employed decision tree data mining technique to develop a classification model for predicting

employees' performance; they claimed that the model or an improved version could assist organisations select the right applicant for a job.

The review of data mining use in predicting employees' performance shows a strong orientation amongst researchers building classification models from employees' past records to gain knowledge of performance patterns and enabling organisations to forecast future performance of employees accurately and to aid in selecting the right candidates for a task.

2.4.2 Prediction of Software Performance

Software developers are keen on discovering the performance of their software in real world. This helps them in making the right decisions about the software and making improvements where necessary (Shu et al, 2009). The truth is that data mining techniques in predicting the performance of software could go a long way in reducing risks and generally benefit software development organisations (Wu et al, 2006).

An important part of software design is testing, this enables developers improve its reliability and clarify design flaws or unintended behaviours not evident during initial design phase (Shu et al, 2009). Data mining techniques in predicting software performance assists in ascertaining faults in the software, allowing software managers to improve the quality of the software, saving time, and cost (Kaur & Sharma, 2018).

Major research on data mining techniques in predicting software performance focus on defect prediction and the technique mostly used is the classification technique.

Chiş (2008) used software metrics in combination with decision tree to predict modules within a software that has defects, the rules acquired from this process can serve as inputs in identifying defects in other software. Pradeep & Abdul (2015) evaluated different classification methods in predicting the reliability of software based on data collected from systems with past failure. Gayatri et al (2009) evaluated different classification methods in constructing a prediction system to detect software defects; they concluded that decision trees generally prove more effective in predicting software faults, however, no algorithm works for every situation and domain specialists must combine different techniques for the best results. Surveys by Karpagavadivu et al (2012); Paramshetti & Phalke (2014); Kaur & Sharma (2018) analysed relevant research work of different data mining techniques used in software fault detection; from these surveys, clustering and classification methods are the major methods used in detecting software fault.

The knowledge gained from the review of data mining techniques in predicting software performance shows that understanding the faults in software design aids in improving the reliability of software. In addition, researchers in this domain mainly use classification or clustering techniques in detecting and predicting software errors.

2.4.3 Prediction of Instructor Performance

Research carried out on predicting performance shows that the prediction of students' academic performance has the highest number of studies (Peña-Ayala, 2014); however, another relevant area of research within the education sector is the study of instructors' performance (Romero et al, 2010). The performance of students and instructors are interrelated (Mardikyan & Badur, 2011). Instructors in this regard are teachers, educators or software that offers some form of instruction for learners during a learning process.

In predicting the performance of instructors, researchers often attempt to compare the relationship between students' performance and instructors' performance, insisting that the performance of instructors originate from the performance of their learners.

Ahmed et al (2016) makes use of four different classification methods to predict instructors' performance based on the evaluation collected from students; their work concludes that students' evaluation of instructors can assist in predicting both the performance of students and instructors. Mardikyan & Badur (2011) attempts to show the factors that affect the performance of instructors from the evaluation of their learners using stepwise regression and classification methods; from the research the most influential factor is the instructors' attitude. Ola & Pallaniappan (2013) proposed a framework using data mining methods for the evaluation of instructors' performance with the idea that if implemented would assist school administrators in decision-making and improve students' academic performance. Agaoglu (2016) used data mining techniques to build seven classification models from students' evaluation of instructors' performance; according to their research, data mining techniques can effectively classify instructors' performance, which can assist instructors improve in their teaching methods and administrators in decision making.

From the review of data mining techniques in predicting instructors' performance, classification is the techniques mostly used to predict instructors' performance. Research in this area studied performance of instructors through evaluation collected from their students.

Another research area where the prediction of performance using data-mining techniques seems to be heading is in the prediction of sports performance (De Marchi, 2011). Like in the research carried out by Arndt & Brefeld (2016) to predict the performance of future soccer players, this research used regression technique to predict the performance of players at the next game based on past events and individual player attributes. From the reviews of performance prediction, the data mining technique most suitable for the task is the classification model, this method used shows the gathering of past data to predict future event.

2.5 Data Mining for Academic Performance

Universities in many countries compete amongst themselves and tend to keep up with latest educational trends as a means of improving the system and keeping the university relevant (Vandamme et al, 2007). The students as the most important resource is at the centre of university concern and their needs considered important. Universities need to know and understand their students to be able to assist them with their needs (Vandamme et al, 2007). In addition, the academic performance of students is at the core of universities concern even as academic reputation of a university is an important indicator in world ranking (The World University Rankings, 2018). With the help of data mining techniques, universities can monitor the performance of students and using machine learning techniques for predicting students' academic performance is already in its adolescence years as a lot of research work done asserts to that fact (Romero & Ventura, 2010).

Research on data mining for academic performance has considered predicting student failure/success rates (Rountree et al, 2004; Winston et al, 2014; Yehuala, 2015), dropout rates (Dekker et al, 2009; Yang et al, 2013; Yukselturk et al, 2014), and predicting academic performance for courses (Al-Saleem et al, 2015; Badr et al, 2016; Bucos & Drăgulescu, 2018).

Academic performance mining research focuses mostly on predicting student performance in web based educational systems and computer based educational system (Daud et al, 2017). This is possible because of the easy access of data within this educational environment. For learning environments where learning is not computer based, acquiring datasets would require researchers to either physically monitor students learning activity, gather data in different formats from different sources, or make use of questionnaires to gather opinions like in the works of Márquez-Vera et al (2013) and Márquez-Vera et al (2013).

The review on predicting students' academic performance using data mining techniques by Shahiri et al (2015), highlights CGPA and internal assessment as the attributes used mostly for predicting students' performance followed by attributes such as demographics, external assessments, extra-curricular activities, previous academic background, and social interaction. Their review also indicated that most researchers predict academic performance using classification methods such as decision trees, naïve Bayes, support vector machine, artificial neural networks, and k-nearest neighbour.

Researchers mining academic performance makes use of several data mining techniques solely or in combination with others. However, the popular data mining techniques used so far are the classification and clustering techniques (Peña-Ayala, 2014).

Some research work has attempted to figure out techniques that perform best in predicting students' academic performance by combining students' learning and personal attributes with previous grades (Romero & Ventura, 2010) and this review discusses few of them. The research by Vandamme et al (2007) classified first year students into three groups of low, medium and high risk as soon as the academic session starts. The work used questionnaires consisting of students' personal history, students' involvement in their academics and students' perception about their academics. This study did not make use of previous academic records and although the research results were not exceptionally great, the discriminant analysis gave the best result out of the three classification techniques used (decision tree, neural networks and discriminant analysis). Kabakchieva (2013) combined features such as students' personal information, pre-university and university records to predict students' academic performance using four classifiers (decision tree, k-nearest neighbour (k-NN), naïve Bayes and rule induction); the results from the research shows that decision tree performs best. Asif et al (2017) used different classification methods to predict final year students' graduation; in their work, they used students' academic records before and after entering university, they concluded that the results were reasonable and they found decision tree algorithm to be the best method.

Research work done in mining academic performance shows that classification methods are best suited for predicting the academic performance of scholars. This thesis focus would be on using the present available features to predict the future performance of students by analysing and mining knowledge from current low performing student and using these attributes to assist future students with same characteristics to perform better.

2.5.1 School Dropout and Poor Academic Performance

Several factors can influence students to drop out of school such as ethnic, social, cultural, family, psychological profiles and academic progress (Aloise-Young & Chavez, 2002). Doll et al, (2013) analysed factors that influence school dropout into push, pull, or fall out. According to the study, pushing out of school are factors within the school that adversely affect students and cause them to leave such as school policies or result of poor behaviour. Pulling out to school are factors such as finance, family or health problems outside the school that distracts them and cause them to leave, while falling out factors relates to poor academic progress. It is obvious that there is a correlation between poor academic performance and school dropout. Most students that dropout due to poor academic performance consider the course too difficult or their grades too poor to continue (Stinebrickner & Stinebrickner, 2014).

Students dropping out of schools have economic effects on the society and individual effects on the lives of school dropouts (Latif et al, 2015). While schools and government find ways to curb the dropout rates in their respective schools and countries, it is imperative to focus on finding ways to improve students' academic performance, as this would go a long way in reducing dropout rates.

The use of data mining in predicting dropout rates in universities has been the concern of many researchers in the educational data-mining domain. Dekker et al (2009) research on predicting dropout rates considered three datasets, which are pre-university data, university data and both attribute sets using different classification methods. The dataset with both attributes of pre-university and university data performed better than the other datasets. In predicting dropout students in an online educational program, Yukselturk et al (2014) combined questionnaire response of students and their continuation of the program to predict dropout. This study compared the performance of four classification methods, which are k-NN, naïve Bayes, decision tree and artificial neural networks, their results recommended the artificial neural networks and decision tree as the best classifiers in student dropout predictions. Rovira et al (2017) developed a predictive model using five different classifiers for predicting students' grades and dropout tendency. Their research also developed a visualization tool to enable tutors understand and interpret the results better.

2.6 Causes of Poor Academic Performance in Developing Countries

Poor academic performance has been a cause of concern for institutions in different countries (Al-Zoubi & Younes, 2015). The causes of academic performance vary from society to society and

from individual to individual. In some institutions/countries especially in developed nations, students' low performance usually relates to personal problems such as emotional trauma or lack of motivation (Banerjee, 2016). However, in many developing countries, institutions and the government also share in the causes of poor academic performance.

- Individual causes: These factors directly relate to students' ability to focus and concentrate on their academic work. Al-Zoubi & Younes, (2015) outlined lack of motivation, fear of failure, students' perception about the course, poor planning, lack of self-confidence, and anxiety about exams as some factors that influence the performance of students. Alami, (2016) mentioned lack of plans for the future, cheating, lack of interest in course and laziness as individual causes of low performance.
- Institutional causes: Some institutional causes of poor academic performance in developing countries that adversely affect students' performance are lack of a conducive learning environment, teachers' lack of required modern educational and psychological knowledge.
- Government causes: Many public institutions in developing countries face the problem of low funding and this creates a poor learning environment, which affects the performance of students (Glewwe & Kremer, 2006). The government has a major role to play in ensuring universities are up to modern standards by creating and enforcing the right policies and making funds available to enhance learning.
- Other causes: These factors affect learners that are generally beyond their control. For example, factors such as family financial background, medical and psychological problems can adversely affect the performance of students (Al-Zoubi & Younes, 2015). Universities need to keep this in mind and create an environment where students with such conditions receive special treatments to boost their performance.

2.6.1 A Focus in Nigeria

While looking at the causes of poor academic performance, it was established the causes of low performance varies from society to society; hence, there are factors inherent in Nigeria that limits the performance of students; for example, the high rate of unemployed graduates creates panic in undergraduate students, as they are uncertain of their future (Okubanjo, 2008; Longe, 2017). When undergraduate students are aware of the amount of unemployed graduates in the country, they tend to lose focus and lack motivation to do well in their studies. Finance related problem is also another

issue in Nigerian higher degree education due to the large amount of low-income earners in the country (Olotu et al, 2015). Also, students that fend for themselves through engagement in temporary jobs or personal trades tend to value their source of income more than acquiring good grades; since in their opinion it is better to focus on their finance than making good grades with very little career opportunities in the future (Nnamani et al, 2014). Another issue is the lack of bursary scheme available in universities and scholarships are very competitive and might only be available to second year students with very good grades; hence, students with poor grades lack support and motivation to do better (Eno-Abasi et al, 2018).

The causes of poor academic performance in Nigeria also distributes amongst individual, institutional and government factors.

- Individual factor: Individual student might lack enthusiasm or the right amount of support to perform well in their academics. Individual factors highlighted by Adeyemi & Adeyemi, (2014b) include students' lack of interest in their course, poor planning and study habit, negative peer influence, students' perception of course as difficult or uninteresting, no support from parents/guardian, family crisis, and students' family educational background. All these factors directly and indirectly affect the performance of students.
- Institutional factors: Many institutions in Nigeria lack the modern technological needs for 21st century learners. The learning environments make it difficult for students to access their full potential as the environment adversely affects their ability to perform well. Some institutional factors that affect Nigerian higher institutions outlined by Adeyemi & Adeyemi, (2014a) are student to teacher ratio, lecturers' interest and commitment, instructors' knowledge of subject and passion for teaching, effectiveness of teaching method, school leadership, school calendar stability, poor school environment, poor teaching materials, lack of adequate educational infrastructure, and poor library facilities.
- Government factors: The lack of stability in many public schools are due to disagreements between the government and the institutions which results in regular strike activities embarked on by academic staff (Ugar, 2018). The government support given to the education sector is also not encouraging, as percentage of national budget for educational purposes is very low (Ani, 2017). Other government related factors include inadequate stable power supply within and outside school environment, poor security in most school

environment, and high rate of poverty in the country affecting the financial state of students' sponsors (Ani, 2017).

2.7 Academic Performance Prediction Modelling

When modelling students' academic performance, researchers have focused on combining different data mining techniques to identify the best technique based on the dataset and developing useful models for future purposes. An important part of model development is feature selection; the right features can make all the difference in system modelling (Strecht et al, 2015). A feature selection technique helps to identify attributes relevant to a data-mining task within a dataset (Beniwal & Arora, 2012.). Selecting the best features for a model includes establishing all subsets of the attributes and evaluating each one (Ramaswami & Bhaskaran, 2009). The feature selection methods used in data mining are filter, wrapper and embedded methods. The filter method ranks relevant attributes based on the overall characteristics of the training data while the wrapper method uses an algorithm to evaluate the accuracy of features in prediction and the embedded method combines the characteristics of both the filter and wrapper methods (Pitt & Nayak, 2007).

From research carried out in modelling students' academic performance, some relevant attributes include previous academic records (Ogor, 2007; Borkar & Rajeswari, 2014), demographic information (García & Mora, 2011; Acharya & Sinha, 2014), parent(s) educational background (Mirashrafi, 2013; Amrieh et al, 2016), parent(s) financial status (Baradwaj & Pal, 2012; David & Gómez, 2014), among others.

Student models developed within EDM domain focus on providing students and tutors with timely information to assist students perform better and enable tutors detect struggling students as early as possible. Kabakchieva, (2013) developed models to predict students' performance, combining students' attributes such as personal, pre-university and university academic performance using classification methods. ElGamal, (2013) developed a model for predicting student academic performance in programming courses using a combination of personal information and previous academic knowledge in programming and mathematics, their results indicate the importance of good grades in mathematics and experience in programming, the model obtained from their research can assist lecturers in providing the necessary support to new students. Romero et al (2008) combined different data mining methods to classify students based on their Moodle usage data and the final scores for different programmes. They achieved this using a data mining tool they developed for this purpose, the model from their research results assists tutors detect students

with learning problems as early as possible and consequently intervening to enhance such performance.

2.8 Chapter Summary and Lessons learnt

This chapter interrogated educational data mining, its application, methods, benefits, challenges and future prospects. It also examined data mining techniques in predicting performance in different areas and highlighted the accomplishments of data mining and its capabilities in understanding and improving these areas. This chapter also extensively discussed the use of data mining in education, specifically in predicting academic performance, which showed the different ways data mining builds education and the society through the development of methods that scholars and educators could deploy in order to improve on their academic performance and tutoring skills respectively. One lesson learned from this chapter is that the common data mining methods used in predicting student performance are clustering and classification methods. The research objectives of this thesis highlight the use of the classification method in identifying relevant features of low performing students and classifying them into different failure groups. Finally, this research aligns with the goal of EDM, which requires collective research in every learning environment in every country to boost the research area and acquire working models for easy implementation across every educational system.

CHAPTER THREE: RESEARCH METHODOLOGY

3.1 Introduction

The goal of this study is to design a system that classifies low performing undergraduate students in NDU using machine-learning techniques. To achieve this, the study follows the CRISP-DM (Cross-Industry Standard Process for Data Mining). The CRISP-DM process is a popular methodology followed in EDM, which consists of six phases outlining steps required for successful knowledge mining.

This chapter describes the CRISP-DM phases and its implementation in this study. In depth, the chapter scans the process of collecting and collating data, the preparation of data for mining, discussion of the five machine learning algorithms selected for the modelling process, techniques for feature selection and techniques for evaluation of the models.

3.2 Educational Data Mining Process

A common methodology followed in the discovery of knowledge is the CRISP-DM process (Chalaris et al, 2014; Oreski et al, 2017); the CRISP-DM process is a well-known data mining process that shows clear paths to achieve project goals (Wirth & Hipp, 2000). The CRISP-DM process depicted in Fig 3.1 has six phases that carry through a project for successful knowledge mining. An overview of the CRISP-DM process provides a good understanding of the process and its implementation in this research. A brief discussion of the six phases of the CRISP-DM model and its use in this research follows:

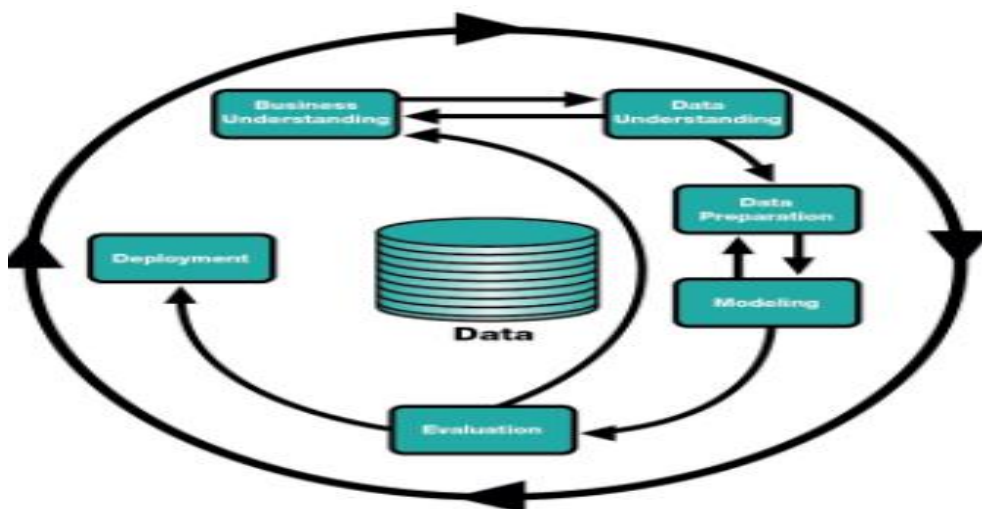


Fig 3.1: The CRISP-DM Process (Olson & Delen, 2008)

Business Understanding: This phase involves setting out research objectives by ascertaining important stakeholders for the research and gathering valuable information to ensure the objectives of the research are attainable.

Data Understanding: This phase involves collecting and examining relevant data. It includes validating the data to rid it of redundant and incomplete values. This phase enables the analysis of the data quality in terms of research objectives and highlights useful patterns in the data.

Data Preparation: This phase handles cleaning (missing or incomplete data) and transforming (converting to suitable format) the collected data; this ensures that the data is appropriate for the selected modelling tool.

Modelling: This phase involves choosing modelling algorithms and applying them to the prepared data to generate new knowledge. This study employs the WEKA modelling tool, which has several classification algorithms to model the data. However, before modelling commences, it is necessary to split the dataset into two parts for training and testing. The splitting helps eliminate bias and analyses how well the model generalizes.

Evaluation: The evaluation phase interprets the results from the modelling phase by locating interesting patterns in the developed models and ensuring that the results meet the objectives of the project.

Deployment: This final stage presents the knowledge discovered from the data mining process by either designing a system or incorporating it into an already existing system. This deployment stage offers stakeholders with the knowledge they need to make better decisions for the organization.

3.3 Framework

This research adopted the six CRISP-DM phases to investigate the problem of low academic performance in NDU. The diagram in Fig 3.2 illustrates the framework followed in this research to achieve the research goals and the discussions of the steps follows.

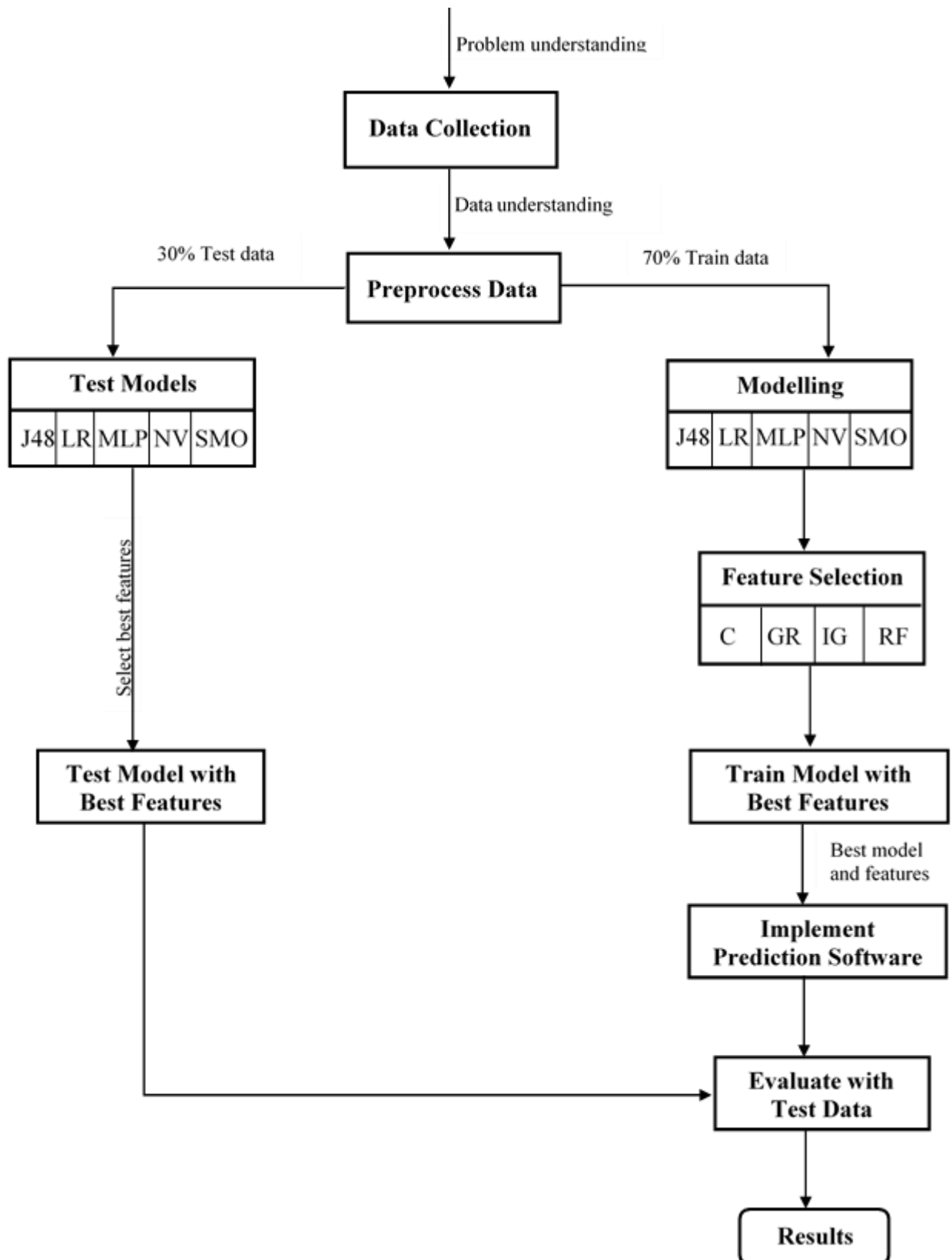


Fig 3.2: The framework of this research

3.3.1 Domain understanding: poor academic performance

This phase involves setting out research objectives by ascertaining important stakeholders for the research and gathering valuable information to ensure the objectives of the research are attainable.

The research objectives stated in Chapter One of this research work describe the goals of the research. Understanding the domain constitutes the first phase of the process and locating the problem of low performance in Niger Delta University undergraduate students within the specific goal of developing a predictive model that classifies the failure level of a new student. This problem requires the use of classification technique.

Looking at the problem of underperformance in Niger Delta University undergraduate students; the student results collected and a brief discussion with key stakeholders at the university confirmed the existence of the problem in the university. These stakeholders include faculty deans, faculty examination officers and heads of departments.

Understanding and stating clearly that the issue of low performance exists in NDU led to gathering of student data and the next step involves understanding the data.

3.3.2 Data Understanding

With the domain of the problem comprehended, the next phase involves data understanding. This entails getting familiar with the different data types in every attribute and identifying data that requires transformation to make it suitable for modelling. The data source is low performing undergraduate students in Niger Delta University, Bayelsa State, Nigeria.

The Niger Delta University has about 10,000 undergraduate students. The data collected from the university shows about 5631 correctly stored undergraduate students' records and 3481 of this population recorded a cumulative grade point average of less than 3.00 on a 5.00 grade point scale. With this high level of poor performance in the university, the relevance of the study is justified. The study sampled 2348 students from different faculties and levels. The Raosoft sample size calculator (Raosoft, 2004) enabled the research to sample this target number. The Raosoft sample size calculator shows that for a population of 3481, a confidence level of 95% and 5% margin of error, a sample size of 347 and above can generalise the results obtained.

The first step in understanding the data requires the collection of data. A description of the data collection process follows.

3.3.2.1 Data Collection

The diagram in Fig 3.3 shows the complete data collection process followed in this study. From the diagram, the data collection process involves letters distributed to key stakeholders to gain permission to conduct the study, which led to meeting with stakeholders to agree on terms regarding data collection process. Next, the collection of existing data from available sources and the next step was the collation of all the data collected into one format for mining, and cleaning data to rid it of inaccurate or incomplete information. The final step, which is data set for mining, is the product derived from the entire data collection process.

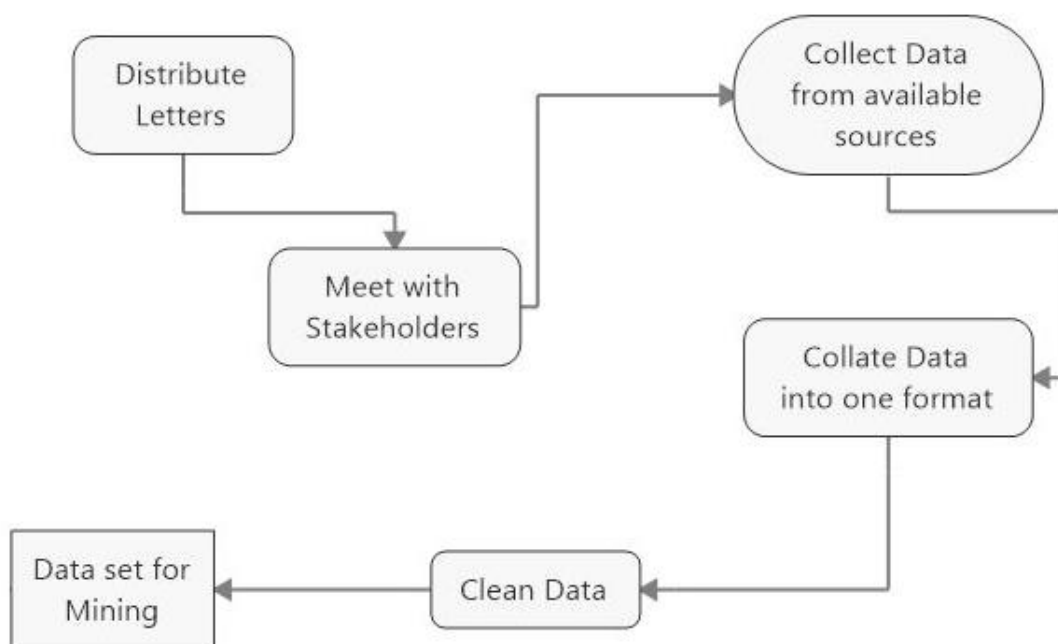


Fig 3.3: The data collection process

Collected Data Records

The first activity involves gathering attributes directly related to underperformance of students through the review of related literature. From the findings of the review, the research identified attributes to focus upon during the data collection activity at the university. Description of the data collection activities follows next.

Activity 1: Literature Review

The review of literature focused on recent and relevant studies about attributes of low performing students and its relationship to school dropout, with a specific focus on the global problem and

then narrowing it down to Nigeria. Through the research on low performing students over the years, scholars acknowledge some attributes as general indicators of the problem. However, other studies confirm that certain attributes are unique to students in different countries and study environment. Section 2.4 in Chapter Two describes the review and findings from this activity.

Activity 2: Secondary Data Collection

Secondary data collection formed the first data collection process in this research. This is necessary to acquire available data stored by the university. The researcher had a meeting with the deputy vice chancellor academics (DVC Acad.) of the university who is in charge of handling all information related to students' academics. The DVC Acad. assisted the researcher by sending out memos to all faculty deans, heads of departments, faculty officers and faculty examination officers to assist the researcher in the data gathering. With all faculty deans and heads of faculty sections notified, the intention of the researcher was to collect data from one faculty and then move to the next faculty. However, the researcher experienced delays from some faculty officers and examination officers and decided to combine two or three faculties depending on the rate of response. Meeting with each faculty officer was mostly productive on the same day with few exceptions where the faculty officers postponed the data collection for a later date. The faculty officers stored the student details in spreadsheet files and gave the researcher the data on USB flash drive. The collection of students' results data involved meeting with examination officers within departments; for some departments where the examination officers were unavailable, the heads of departments provided the data, which were in either PDF or hard copy files. The researcher transferred each file collected through USB flash drive immediately into the researcher's laptop and stored all hard copy files in a file jacket.

At the end of the data collection, the researcher gathered all the files received and stored them in a folder for collation and cleaning.

3.3.3 Data Preparation Process

The secondary data collected from the university had many incomplete and inaccurate data. The student details collected from the university is about ten thousand four hundred and seventy-two (10472) records; after manually inserting CGPA from the PDF or hardcopy, the total number of students with CGPA is five thousand six hundred and thirty-one (5631) records. Three thousand four hundred and eighty-one (3481) records formed students with CGPA less than 3.0, which forms the entire population of low performing undergraduate students in this research.

The process of collating the records obtained from the university required a lot of time. First, the researcher arranged the student data collected from faculty officers into one spreadsheet file with each faculty allocated a sheet of its own to enable easy management of records, then added a new field called CGPA to all faculties. Next, the researcher manually keyed in the CGPA for each student record, which took a lot of time because of the size of the data. The researcher also spent time verifying that each CGPA keyed in tallied with the student data and that the figure was accurate. With the CGPA added to the dataset, the researcher sorted the CGPA field to extract all students with CGPA, moved the set of data from all faculties into a new spreadsheet file, and finally sorted the data to acquire students with CGPA less than 3.0.

Data cleaning considered as an important step in the pre-processing stage of data mining implies the detection and removal of errors and discrepancies from data to improve its quality (Rahm & Do, 2000). To ensure the dataset collected are of high quality, the researcher that all records are complete and within required boundaries and removed all inconsequential fields regarding the research purpose.

The cleaning process proved challenging because of the huge amount of incomplete data. There were several records with either no data or incomplete data, for example, about thirty-three (33) values were omitted for sex field and sixty-nine (69) values were omitted for date of birth field (with about twenty (20) incorrect values for year of birth). These omitted values occurred randomly and some records contained two or more of these missing values. The data provided names of students, which assisted in predicting the sex for these missing records. The missing values for date of birth used average date of birth values of students within the same department and level to predict the average date of birth for these records. The researcher also noted that about fifty-six (56) records had a CGPA of zero (0) and decided to remove these records because there is the possibility that these set of students registered for the courses but did not seat for the examinations; although it is also possible for students to fail all courses.

Hence, the clean secondary dataset collected from the university saved in a spreadsheet file formed the main dataset for mining.

3.3.3.1 Attribute Selection

The attributes selected for mining at the end of the data preparation process are twenty-five, including the class attribute which is CGPA. Table 3.1 shows these selected attribute fields, their variables and corresponding values. From the table, some attributes selected are demographic data,

type of previously attended schools, sponsor information, course interest, etc. The Average SSCE score attribute is the average of students' previous academic performance. Students write a minimum of seven subjects and maximum of nine subjects during the examinations and earn grades A, B, C, D, E or F based on their performance. For the purpose of the research, the values 6, 5, 4, 3, 2 or 1 replaced the respective grades and the sum for each student divided by the number of subjects the student wrote gives the value of the average SSCE score.

Table 3.1: Description of data fields and their respective values.

FIELDS	VARIABLES	VALUES
Sex	M F	Male Female
Age	B30 30A	Below 30 years 30 years and above
Marital status	S M	Single Married
Attended primary school	NO YES	No Yes
Secondary school type	PRI PUB	Private Public
Secondary school area	URB RUR	Urban Rural
Sponsor type	GUAD PAR SELF	Guardian Parents Self-sponsor
Sponsor qualification	DEG NODEG NOEDU	Educated with degree Educated without degree No formal education
Sponsor income	LOW MED HIGH	Below N50,000 N50,000 – N100,000 Above N100,000
Sponsor support	LOW MED HIGH	Little support Average support Great support
Family size	SMALL MED LAR	1 – 4 5 - 9 Above 9
Work and study	YES NO	Yes No
University accommodation	CMPS OFFCMPS	Campus Off-campus
Years before admission	NONE B5 5A	None Below 5 years 5 years and above
Course from Jamb	YES NO	Yes No

Course interest	LOW AVE HIGH	Little interest Average interest High interest
Weekly study time	LOW AVE HIGH	Less than 10hrs 10 – 20hrs Above 20hrs
Postgraduate degree	NO YES NS	No Yes Not sure
Own smart phone	YES NO	Yes No
Smart phone assistance	ASGMT STUDY NONE	For assignment For studying None
Sports activeness	LOW HIGH	A little active Very active
Jamb score	LOW AVE HIGH	Below 180 180 – 250 Above 250
Post-UTME score	LOW AVE HIGH	Below 180 180 – 250 Above 250
Average SSCE score	LOW AVE HIGH	Less than 4.00 4.00 – 4.99 5.00 and above
CGPA	HL LL	2.50 - 2.99 0.01 – 2.49

3.3.4 Modelling

For the modelling phase, this research used the WEKA modelling software. The Waikato Environment for Knowledge Analysis (WEKA) is a collection of machine learning algorithms for data mining tasks, containing tools for data preparation, classification, regression, clustering, association rules mining, and visualization. The modelling task requires the use of classification techniques. The classifier models considered in line with the research objectives are J48 decision tree, logistic regression, multilayer perceptron, naïve Bayes and sequential minimal optimization. These classifiers are popular in data mining research and commonly used in EDM domain. A brief discussion of these algorithms follows.

3.3.4.1 J48 Decision Trees

A decision tree depicts a flowchart-like structure with internal nodes or non-leaf nodes representing tests on attributes and terminal nodes or leaf nodes signifying class labels (Han et al,

2011). The decision tree is a model for prediction where classification of instances takes place following the trial of satisfied conditions from the root of the tree until it reaches a leaf, which ultimately corresponds to a class label (Romero et al, 2008). Converting a decision tree to a set of rules enables the alleviation of the effects of the strict hierarchical structure (Aggarwal, 2015). While building decision trees, it is essential to ensure that the algorithm finds the most optimal tree, and to achieve this, the splitting and stopping criteria have to be known and explicated (Tan et al, 2013). The algorithm commonly makes use of the degree of impurity of child nodes to determine the best splits; impurity measures often used are entropy, Gini index and classification error (Tan et al, 2013). Equations 3.1, 3.2 and 3.3 are the formulas for the impurity measures:

$$Entropy(t) = -\sum_{i=0}^{c-1} p(i|t) \log_2 p(i|t) \tag{Eq. 3.1}$$

$$Gini(t) = 1 - \sum_{i=0}^{c-1} [p(i|t)]^2 \tag{Eq. 3.2}$$

$$Classification\ error(t) = 1 - \max[p(i|t)] \tag{Eq. 3.3}$$

The difference in entropy is termed information gain and the algorithm selects the attribute with highest information gain as best splitting attribute for the node (Han et al, 2011). Subsequently, the gain ratio determines the goodness of a split. The gain ratio is the ratio of information gained to the core information and the algorithm selects the attribute with the maximum gain ratio as the splitting attribute (Han et al, 2011). The Gini index considers all the subsets of an attribute and selects the one with minimum Gini index as the best splitting attribute, it also enforces that the tree split is binary (Han et al, 2011). Fig 3.4 illustrates a simple decision tree, showing the root node, internal nodes and the leaf nodes, which represents the class label.

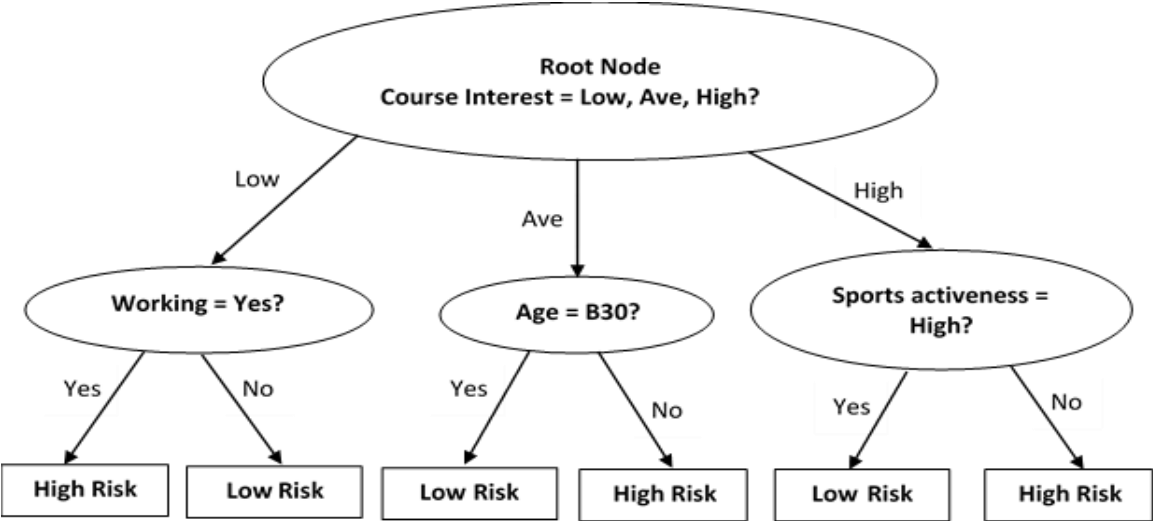


Fig 3.4: A simple decision tree (Larose & Larose, 2014)

The decision tree algorithm has different types and some of the widely used types are Iterative Dichotomiser 3 (ID3), Classification and Regression Tree (CART), Chi-square Automatic Interaction Detection (CHAID) and a successor to the ID3 called C4.5 (Adeyemo et al, 2015). The J48 is the java implementation of C4.5 in WEKA modelling tool; it is so-called because the algorithm implements an upgraded form of the C4.5 algorithm called C4.5 version 8 (Han et al, 2011). The C4.5 algorithm offers more than the ID3 with improvements such as allowing the handling of continuous and discrete features, pruning trees to reduce its size, dealing with dataset that contains missing values and the conversion of the trees into sets of rules (Han et al, 2011; Singh & Gupta, 2014).

3.3.4.2 Logistic Regression

The logistic regression models the probability of an event happening as a set of predictor variables and it is best for tasks with categorical binary class values (Kantardzic, 2011). The main distinction between the popular linear regression and logistic regression is that linear regression model produces its output as a continuous value represented as a straight line on a graph while logistic regression model fits its output as a curve on a graph and gives its result as a dichotomous value (Han et al, 2011). The logistic regression has two model forms called binary logistic regression and multinomial logistic regression (Park, 2013). The binary logistic regression is for tasks with two dependent variable values and the multinomial logistic regression is for tasks with more than two dependent variables values; however, the independent variables for both forms can either be continuous or categorical (Park, 2013).

The logistic regression model fits the task of this study as it considers the classification of low performing students into two groups. This study looks at the dependent variable values, which are HL and LL, and 24 categorical independent. The output of a logistic model ranges from 0 to 1 (Kleinbaum & Klein, 2010) and for the purpose of the logistic regression, the study considered the class HL as 0 and LL as 1 and the 24 independent variables as X_1, X_2, \dots, X_{24} . With the observations of the independent variables, the logistic regression model considers the probability that a student is in either of the two classes, for an output of 0.5 and above the class is 1 and for less than 0.5 the class is 0.

The conditional probability that the output (D) equals 1 considering the independent variables as given in Eq. 3.4 can be obtained from the formula given in Eq. 3.5 according to Kleinbaum & Klein (2010).

$$P(D = 1|X_1, X_2, \dots, X_{24}) \tag{Eq. 3.4}$$

$$P(D = 1) = \frac{1}{1 + e^{-(\alpha + \sum \beta_i X_i)}} \tag{Eq. 3.5}$$

The values of α and β are unknown parameters that are estimated based on data gotten from the values of the independent variables and the dependent variable outcome (Kleinbaum & Klein, 2010).

3.3.4.3 Multilayer Perceptron (MLP)

The multilayer perceptron is a feed forward artificial neural network made up of the input layer, one or more hidden layers of nodes and an output layer of nodes (Kantardzic, 2011). From the definition above, an artificial neural network is termed ‘artificial’ because it attempts to imitate the nervous system of the human brain, which communicates with neurons by sending information in the form of signals through directed connections to each other (Kruse et al, 2016). In addition, it allows the transfer of information in one direction from input, hidden layers to output and uses a technique called back propagation for training (Marsland, 2014). The diagram in Fig 3.5 shows the structure of a multilayer perceptron, which illustrates the input layer feeding information forward into the hidden layers for processing and sending the response to the output layer.

The input layer of an artificial neural network represents the attributes fed into the model while the output layer signifies the labelled class; therefore, for this study the input layer are the 24 features and the output layer is the target class students belongs, which can either be HL or LL.

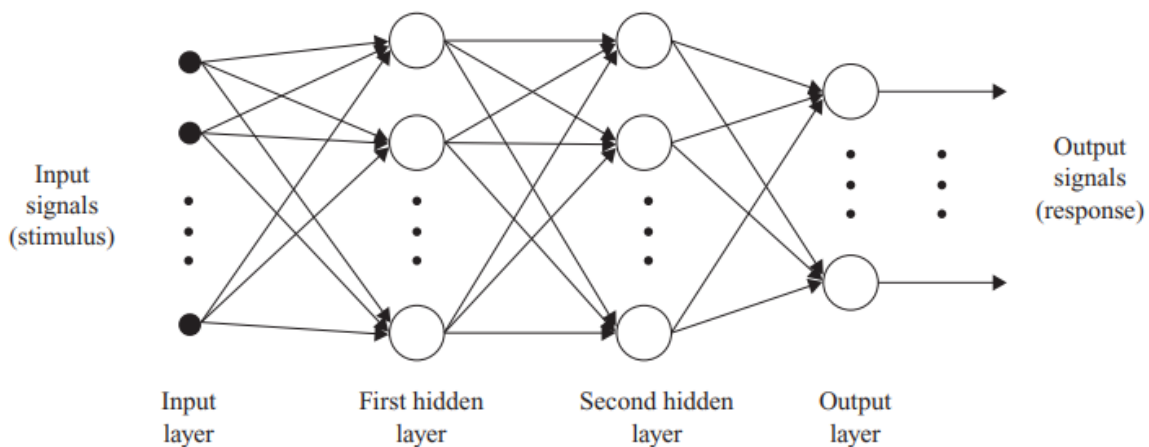


Fig 3.5: Structure of a multilayer perceptron with two hidden layers (Kantardzic, 2011).

The multilayer perceptron model has shown a high level of accuracy in several applications; however, the training time is slow especially with many features, the computations of the hidden layers is difficult to interpret and for the model to perform very well it requires many data for training (Panchal et al, 2011; Asif et al, 2014).

3.3.4.4 Naïve Bayes Bayesian Classifiers

Bayesian classifier presents models for representing probabilistic relationships among multiple interacting variables (Husmeier, 2005; Tan et al, 2013). Classifiers such as the naïve Bayes and the Bayesian belief network model probabilistic relationships between attribute set and class variable (Tan et al, 2013). According to Han et al (2011), the naïve Bayes classifier is similar in performance with other classification algorithms such as decision tree and some neural network classifiers. In addition, the naïve Bayes employs the Bayes theorem for conditional probabilities and assumes that the attributes are conditionally independent (Tan et al, 2013; Aggarwal, 2015). This classifier is robust to remote noise points and irrelevant attributes; however, correlated attributes reduce its performance because the assumption of conditional independence no longer holds (Tan et al, 2013).

The naïve Bayes classifier derives its formula from the popular Bayes theorem depicted in Eq. 3.6 and the Eq. 3.7 represents the formula for the naïve Bayes Classifier (Tan et al, 2013)

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \dots\dots\dots \text{Eq. 3.6}$$

$$P(A|B = b) = \prod_{i=1}^n P(A_i |B = b) \dots\dots\dots \text{Eq. 3.7}$$

From the Eq. 3.7 the value *A* represents the feature set with *n* features {*A* = *A*₁, *A*₂, ..., *A*_{*n*}} and for this study, the *n* features equals the 24 features collected.

3.3.4.5 Sequential Minimal Optimization (SMO)

The support vector machine (SVM) is a supervised learning model used for classification tasks by searching for the hyperplane that gets the most out of the margin that exists between two classes (Suthaharan, 2016). This algorithm is for classification tasks where the target class is dichotomous (Hsu & Lin, 2002). However, the traditional techniques used for training with the SVM model are not fit for problems with large size (Zeng et al, 2008). Therefore, to improve training efficiency, the sequential minimal optimization (SMO) algorithm provides solutions to the quadratic

programming problems that arises in the course of training in SVM (Platt, 1998). The SMO is popular because it is simple and performs faster than other SVM training algorithms (Zeng et al, 2008). To show a pictorial image of the SVM, Fig 3.6 shows the SVM with a hyperplane separating the two classes used for this study.

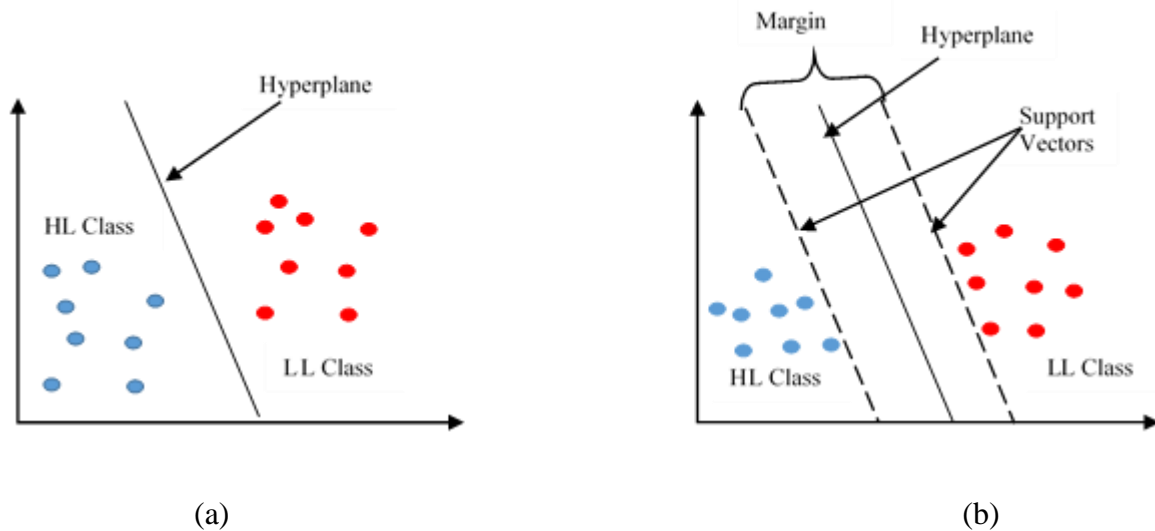


Fig 3.6: The support vector machine showing (a) the separation of the HL class and the LL class with a hyperplane and (b) the point with the highest margin.

From the diagram, the SVM selects the largest margin between the HL class and the LL class; the strength of the SVM model comes from the ability to separate both classes with the largest boundary.

3.3.4.6 Feature Selection Techniques

Data mining typically works with a large amount of data that contains many features and over time as the size of data increases, it creates more features (Dash & Liu, 1997; Saeys et al, 2007; Hira & Gillies, 2015). When a dataset has many features, it may contain some irrelevant features; meaning features that do not contribute much value to the model designed. Hence, it is good practice to identify and select the most important features within the dataset and the technique used in selecting relevant features termed 'feature selection' aids in removing redundant features from the dataset (Hira & Gillies, 2015). This works by selecting important features from the dataset that retains the value of the dataset and produces good results from the modelling process (Chandrashekar & Sahin, 2014). Saeys et al (2007) outlines the following feature selection properties: overfitting and increasing the model performance, offering faster and inexpensive models and to gain a better understanding of the methods described by the data.

This feature selection process generally follows four steps in evaluating and validating the best features in a dataset and Fig 3.7 depicts these steps. From the diagram, the complete dataset goes through generating a subset of features, evaluates these features to determine its relevance, checks with predefined stopping criteria and when it is met, outputs the selected features for validation (Dash & Liu, 1997).

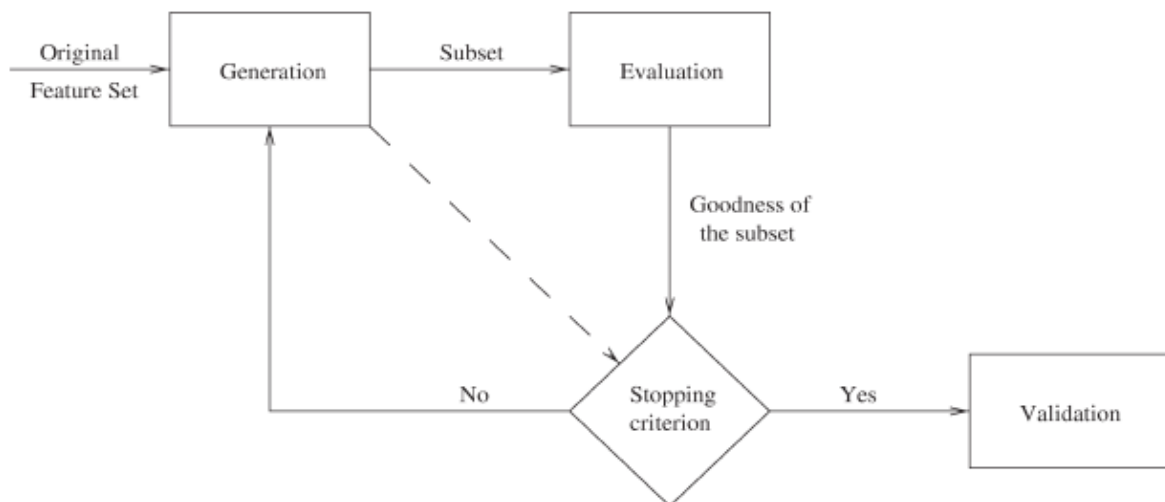


Fig 3.7: General feature selection process (Dash & Liu, 1997)

Some feature selection techniques used in classification methods are the filter, wrapper and embedded methods. The discussion on these techniques follows.

Filter: The filter method works by ranking features in order of importance or usefulness (Shardlow, 2016). It is carried out as a pre-processing step independent of classifiers and has the benefit of being fast (Saeys et al, 2008). Concisely the filter method selects and lists out the best features discovered from the dataset prior to the classification task. Some examples of the filter technique include correlation based feature selection, ReliefF, information gain, fast correlation based feature selection, gain ratio, Markov blanket filter (Sánchez-Marño, et al, 2007; Mwadulo 2016; Shardlow, 2016)

This study employs four filter methods incorporated in WEKA namely correlation, gain ratio, information gain and ReliefF for selecting the best features from the dataset used in this study.

Wrapper: The wrapper method searches for the best set of features by working with a classifier model. It achieves this by using a set of features with the model and makes a decision to keep or discard features from the selected set based on the acquired results (Das, 2001). It is termed “wrapper” because the attribute selection process wraps itself around the classification model

(Chang et al, 2013). This method displays a higher level of accuracy compared to the filter method; however, it can lead to overfitting, it is slow and it is computationally demanding (Saeys et al, 2007). Some examples of wrapper methods include greedy forward search and exhaustive search (Shardlow, 2016).

Embedded: The embedded method of feature selection combines the characteristics of both filter and wrapper methods to offer a balance between performance and computational cost (Saeys et al, 2008). This method works with classifier models at a lesser computational cost and implements in such a way that the in-built feature selection works by reducing the features (Mwadulo 2016; Hameed et al, 2018). Some examples of embedded methods include LASSO and RIDGE regression (Hameed et al, 2018).

3.3.5 Predictive System Methodology

To carry out the design and implementation of any software, it is required to follow a software development methodology. A software development methodology helps in planning and monitoring the process of building information systems (Segue Technologies, 2015). The use of a software methodology enables the smooth cycle of the software development from start to finish. In line with this, this research followed the rapid prototyping methodology in developing the predictive software that is detailed in the following segment.

3.3.5.1 Rapid prototyping

Rapid prototyping methodology strives to design the prototype of a software using less effort compared to the production and implementation of a software for operational use (Devadiga, 2017). The logic is to get a working software early enough to allow feedback and analysis with clients during the software development process (Kordon, 2002).

A prototype is a working model of a system that shows a selected part of the system properties such as the design layout, response times or calculated outputs (Devadiga, 2017). This prototype enables clients to have a visual picture of the working software early enough in the software development process and creates room for feedback and improvement on the software before the development and implementation of the final product.



Fig 3.8: Rapid Application Development model (Kumar & Bhatia, 2014)

The rapid application development model follows four processes as depicted in Fig 3.8. From the diagram, the first step is requirement planning, the next step is user description followed by the penultimate construction and the ultimate cutover.

The requirement planning involves studying the problem, gathering requirements and deciding on the requirements for the project. At the end of this stage, the project team assesses the objectives and prospects of the project and decides on the basic requirements. The second step develops the system based on the user description or expectation using simple prototype methods like sketching the system or designing of storyboard. This stage enables the client and project team to agree on the design and appearance of the software.

The construction step involves transforming the system prototype developed into a working model. This phase comprises preparation for construction of the system, coding, software development and testing. At the end of this step, the design team produces a complete working model of the software.

The last step, which is the cutover stage, is for implementation of the system. This step encompasses launching of the system for use and includes testing, debugging, data conversion and generally making the system fit for use.

The rapid prototyping methodology fits this research as it presents a working software early once the requirements are known; and this helps in providing a working product early enough. The goal of designing a software in this research provides an easy way for users of the system to predict student performance without repeatedly making use of data mining software. Based on this goal, it is important to quickly design a software that makes this possible and make changes on the software in time as required.

3.3.6 Evaluation

Methods of evaluating models in machine learning include splitting the data and K-fold. Splitting the data involves dividing the dataset into two, the first part (training sample) is used to the train

the algorithm and the second part (test or validation sample) is used to evaluate the performance of the algorithm (Arlot, & Celisse, 2010). The K-Fold experimental design is the evaluation method commonly used in machine learning; this method combines the dataset for both training and testing (Anguita et al, 2009). The 10-fold evaluation design splits the dataset randomly in 10 parts and builds the model by training and testing the model 10 times. The training process involves using 90% of the dataset for training and 10% of the dataset for testing on each iteration. Experiment results given at the end of the process is the confusion matrix. A confusion matrix is the classification performance summary of a classifier with regard to some given test data (Ting, 2017).

This study makes use of the WEKA software and some metric measures assisted in determining the performance of the classifier models, a brief discussion of these metric measures follows.

3.3.6.1 Metrics of Evaluation in WEKA

The evaluation process in this research makes use of 10-fold cross-validation to measure the performance of the classifier models. Generally used metric measures in literature for performance evaluation and available in WEKA are specificity, sensitivity, precision, F-Measure (Shaikh et al., 2015), Cohen’s Kappa (Romero and Ventura, 2010), Receiver Operating Characteristics (ROC) area (Sarlis and Christopoulos, 2014) and Root Mean Squared Error (Pardos et al., 2012).

The confusion matrix as earlier stated summarises the classification performance of classifiers models and the results aids in analysing the accuracy of developed models. Fig 3.8 shows the diagrammatical representation of a confusion matrix.

		Classifier Prediction	
		Positive	Negative
Actual Value	Positive	True LL	False LL
	Negative	False HL	True HL

Fig 3.9: A Confusion matrix

Prevalence

Prevalence is a basic measure of evaluation derived from the confusion matrix. It measures the ratio of the actual positives to the entire dataset (Hripcsak, 2012). Prevalence is an important measure as it indicates the target class distribution of the dataset under study. Eq. 3.8 shows the formula for prevalence.

$$Prevalence = \frac{True\ LL + False\ LL}{True\ LL + False\ HL + False\ LL + True\ HL} \dots\dots\dots Eq. 3.8$$

Recall or Sensitivity

Recall, also called sensitivity, measures the ability of a model to classify the positives efficiently (Sokolova & Lapalme, 2009). It is the ratio of correctly identified number of True Positive records from the actual Positive records in the data and it measures the proportion of actual positives correctly identified. Eq. 3.9 shows the formula for sensitivity.

$$Recall = \frac{True\ LL}{True\ LL + False\ LL} \dots\dots\dots Eq. 3.9$$

Specificity

Specificity measures the ability of a model to classify the negatives efficiently (Sokolova & Lapalme, 2009). It is the ratio of correctly identified True Negative records from the actual Negative records in the data and it measures the proportion of negatives correctly identified. Eq. 3.10 shows the formula for specificity.

$$Specificity = \frac{True\ HL}{False\ HL + True\ HL} \dots\dots\dots Eq. 3.10$$

The specificity and sensitivity metrics classifies True Positives and True Negatives of a dataset; therefore, a perfect classifier would be 100% sensitive and 100% specific, which means the classifier identified all the records correctly.

Precision

Precision also called positive predictive value (PPV) measures the predicted positive values of the model, it is the ratio of correctly predicted True Positives to all the Positive values predicted by the model (Hripcsak, 2012; Nisbet et al, 2017). Eq. 3.11 shows the formula for precision.

$$Precision = \frac{True\ LL}{True\ LL + False\ HL} \dots\dots\dots Eq. 3.11$$

F-Measure

F-Measure determines the efficiency of the classifier’s prediction of the target class by combining both precision and recall to attain a balanced average value (Shaikh et al., 2015). It is the harmonic mean of the model’s precision and recall values (Sasaki, 2007). The Eq. 3.12 shows the formula for F-Measure.

$$F - Measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \dots\dots\dots Eq. 3.12$$

Receiver Operating Characteristics (ROC) Area

The ROC metric determines the ability of a model to avoid misclassifications through a graph by plotting sensitivity against specificity (Sokolova & Lapalme, 2009; Hripcsak, 2012). It is a useful and popular metric used to analyse model performance with applications in different fields such as medicine and meteorology (Sarlis & Christopoulos, 2014). The ROC metric is preferred because it achieves a balance between sensitivity and specificity and a model with high sensitivity value and low specificity value results in producing a larger ROC area, which means that the model achieves high accuracy in identifying the True Positives.

Matthews Correlation Coefficient (MCC)

The Matthews Correlation Coefficient (MCC) is a metric measure for binary classification that is suitable for imbalanced dataset that happens when the sample size of data classes is unequally distributed, that is when one data class has a lot more records than the other class (Boughorbel et al, 2017). The MCC results lies between -1 to +1, where -1 indicates total disagreement, 0 indicates random predictions and +1 indicates total prediction (Baldi et al, 200). The equation (Eq. 3.13) shows the formula to calculate the MCC directly from the confusion matrix.

$$MCC = \frac{True\ LL \times True\ HL - False\ HL \times False\ LL}{\sqrt{(True\ LL + False\ LL)(True\ LL + False\ HL)(True\ HL + False\ HL)(True\ HL + False\ LL)}} \dots\dots Eq. 3.13$$

Precision-Recall Curve (PRC) Area

The PRC area is a metric measure obtained by plotting the model’s values of precision against recall (Srivastava & Singh, 2015). It is suitable for imbalanced data as it offers the performance of

a model by considering only the values of Positives and the difference between PRC and ROC is that PRC does not consider the values of Negatives (Saito & Rehmsmeier, 2015).

Cohen's Kappa

The Cohen’s Kappa or Kappa statistics measures the inter-rater reliability of categorical objects; the inter-rater reliability is the level of agreement between two binary raters (Wood, 2007). This metric measure produces values between -1 to +1, where a value close to +1 signifies agreement between the raters and a negative value depicts disagreement between the raters, however, the kappa has a benchmark value of 0.60; therefore, a model that produces a score below the benchmark has low reliability (McHugh, 2012). The equation (Eq. 3.14) states the formula for the Cohen’s Kappa (McHugh, 2012).

$$\kappa = \frac{\text{Pr}(a) - \text{Pr}(e)}{1 - \text{Pr}(e)} \dots\dots\dots \text{Eq. 3.14}$$

From the formula, κ represents the Cohen’s Kappa value, $\text{Pr}(a)$ signifies the actual observed agreement and $\text{Pr}(e)$ signifies the proposed probability of chance agreement.

Mean Absolute Error (MAE)

The mean absolute error is a valuable and popular metric for model evaluation (Chai & Draxler, 2014). The mean absolute error measures the proximity between predictions and ultimate results of two continuous variables (Willmott & Matsuura, 2005) and the formula in Eq. 3.15 shows the formula for the mean absolute error (Wang & Lu, 2018).

$$MAE = \frac{1}{n} \sum_{i=1}^n |P_i + A_i| \dots\dots\dots \text{Eq. 3.15}$$

From the formula, n represents the number of samples, P_i is the predicted value and A_i is the true value.

Root Mean Squared Error (RMSE)

The RMSE metric evaluates classifier performance with applications in fields such as climate research, meteorology and air quality (Chai & Draxler, 2014). Pardos et al (2012) applied it in educational data mining research to measure the performance of several classifiers based on the classifiers’ size of error. This metric of error evaluation is the square root of the mean squared error obtained from the equation in Eq. 3.16 (Wang & Lu, 2018) below.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (P_i - A_i)^2} \dots\dots\dots \text{Eq. 3.16}$$

From the formula, n represents the number of samples, P_i is the predicted value and A_i is the true value.

Relative Absolute Error (RAE)

The relative absolute error measures the performance of models by obtaining the ratio of the mean absolute value of actual predicted errors and the mean absolute value of the models’ predicted errors (Guo et al, 2015). A relative absolute error can range from zero to infinite value; however, a good model should have value close to zero. Eq. 3.17 shows the formula for the relative absolute error (Botchkarev, 2018).

$$RAE = \sum_{i=1}^n \frac{|P_i - A_i|}{|\bar{A} - A_i|} \dots\dots\dots \text{Eq. 3.17}$$

From the formula, P_i represents the predicted value, A_i represents the actual value, and \bar{A} represents the mean of the actual values over the training data.

Root Relative Squared Error (RRSE)

The root relative squared error measures the performance of models by obtaining the squared root for the ratio of the mean square value of actual predicted errors and the mean square value of the model’s predicted errors (Guo et al, 2015). For a classifier, the smaller the root relative square error value, the better the performance of the classifier. Eq. 3.18 shows the formula for the relative absolute error (Botchkarev, 2018).

$$RRSE = \sqrt{\sum_{i=1}^n \frac{(P_i - A_i)^2}{(\bar{A} - A_i)^2}} \dots\dots\dots \text{Eq. 3.18}$$

From the formula, P_i represents the predicted value, A_i represents the actual value and \bar{A} represents the mean of the actual values over the training data.

3.4 Chapter Summary

This chapter commenced with a brief overview of the educational data mining process using the CRISP-DM methodology. It also reflected on the research framework utilised in this study with discussion on the process followed, which are domain understanding, data understanding (data

collection), data preparation (attribute selection), data modelling and evaluation. For the modelling and evaluation parts, the chapter briefly explores the classifier models used in the research and the metric measures for evaluating the models.

In conclusion, this chapter discussed the overall machine learning process followed in this study to achieve the research objective.

The next chapter presents the machine learning process followed, the results obtained from the process and discussion of the results.

CHAPTER FOUR: DATA MODELLING, RESULTS AND DISCUSSIONS

4.1 Introduction

The main aim of this research is to develop a predictive application that classifies low performing undergraduate students in NDU into two groups called HL and LL. To achieve this aim, some machine learning (ML) algorithms utilized on the dataset collected in NDU help to determine the ML algorithm that best models the data and subsequently utilises that algorithm to develop the predictive system for NDU. This chapter presents the results and discussion emanating from the educational data mining process using the WEKA modelling tool. In Chapter Three, this study described the entire methodological approach followed, involving data gathering, collation and discussions on chosen algorithms and metrics for evaluation.

4.2 Presentation and Discussions of Results

In machine learning, a model can generalise the training data very well and behave poorly on new unseen data; this problem is called overfitting (Hämäläinen & Vinni, 2010). One method used to identify overfitting is the splitting of dataset into two parts, one part for training and the second part for testing (Ballard et al, 2007). The splitting method used in this research is 70/30; the first part with 70% of the dataset forms the dataset for training and the second part with 30% forms the dataset for testing how well the model generalises on unseen data.

In preparation for mining, this research loads the complete dataset saved in Microsoft Excel csv file into the WEKA modelling tool for mining. Weka enables the automatic conversion of csv to arff and it is necessary to covert to arff as WEKA can only work with test files saved as arff.

In pre-processing the data, the research used the WEKA modelling tool to convert the csv file into arff file and then made use of the resample filter to split the data into two parts for training and testing. The resample filter enables the splitting of the data into two parts without duplicating so there is certainty that the model has not seen any of the test data before. The algorithm below describes the steps followed to resample the data.

Algorithm 4.1: Resampling of dataset

- Step 1. Start
- Step 2. Open WEKA modelling software
- Step 3. Select Explorer menu

- Step 4. Select Open file
- Step 5. Choose dataset file from saved location
- Step 6. Click Open
- Step 7. Dataset file is loaded to WEKA
- Step 8. In the Pre-process menu, choose filter type, select unsupervised, select instance and select resample
- Step 9. For the 70% training dataset
 - a. Click on the resample filter to see the properties
 - b. On the properties menu, change the SampleSizePercent to 70
 - c. Change noReplacement to True
 - d. Click OK
 - e. Click Apply
 - f. Then click Save to save the training dataset
- Step 10. For the 30% test dataset
 - a. Begin by clicking undo to get the entire dataset
 - b. Then click on the resample filter to see the properties
 - c. Keep the noReplacement option as True
 - d. Change the invertSelection option to true (this ensures that only the remaining 30% dataset is selected)
 - e. Click OK
 - f. Click Apply
 - g. Then click Save to save the test dataset
- Step 11. Stop

With both training and test dataset in separate files, the mining process begins with running the analysis for the training dataset and making use of the model built for testing of the unseen test data. The modelling techniques used for classification in this research are J48 decision tree, logistic regression, multilayer perceptron, naïve Bayes and sequential minimal optimization algorithms.

4.2.1 Presentation and Interpretation of Training Dataset

For all the algorithms used, the best model is the one that has the best values for the selected metrics of performance measure. The metrics selected for this research are Kappa statistics, RMSE, Recall, Specificity, F-Measure and ROC area. The performance of the models was measured using the 10-fold cross validation method in WEKA. The presentation of results and

discussions of the five algorithms used in this study, which include J48, LR, MLP, NV and SMO are presented next.

The J48 Decision Tree

The J48 classifier is the first model built in this research through the use of WEKA modelling tool. The J48 algorithm is an extension of the C4.5 decision classifier's eighth version in Java that provides more capabilities than the C4.5 algorithm (Han et al, 2011). This classifier builds a tree to show classification and it is a popular method used for classification tasks in EDM (Hussain et al, 2018). Table 4.1 summarises the results obtained from the modelling process with the use of the J48 algorithm and the diagram in Fig 4.1 depicts the performance of the model in WEKA environment.

Table 4.1: The summary of training dataset results obtained from the J48 classifier model

Metrics	Value
Kappa statistic	0.8979
Mean absolute error	0.0643
Root mean squared error	0.2085
Relative absolute error	13.8717%
Root relative squared error	43.299%
Recall or Sensitivity	0.977
Specificity	0.912
Precision	0.951
ROC Area	0.961
F-Measure	0.964
MCC	0.899
PRC Area	0.960

```

Time taken to build model: 0.07 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      1566           95.3135 %
Incorrectly Classified Instances     77             4.6865 %
Kappa statistic                     0.8979
Mean absolute error                  0.0643
Root mean squared error              0.2085
Relative absolute error              13.8717 %
Root relative squared error          43.299 %
Total Number of Instances           1643

=== Detailed Accuracy By Class ===

                TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                -----  -----  -
                0.912   0.023   0.958     0.912   0.934     0.899   0.961    0.941    HL
                0.977   0.088   0.951     0.977   0.964     0.899   0.961    0.960    LL
Weighted Avg.   0.953   0.064   0.953     0.953   0.953     0.899   0.961    0.953

=== Confusion Matrix ===

  a    b  <-- classified as
547  53 |   a = HL
 24 1019 |   b = LL

```

Fig 4.1: The J48 classifier model showing the performance of the training dataset

The Logistic Regression Classifier

The logistic regression classifier, which is a binary classifier model is the second model built in the study. The logistic regression model looks at the probability that a student has a high risk of performing poorly based on the attributes considered as predictors for students in that group. The logistic regression model is a popular method used by several researchers in EDM community for classification task (Peña-Ayala, 2014).

Table 4.2 summarises the results obtained from the modelling process using the logistic regression algorithm and the diagram in Fig 4.2 portrays the performance of the model in WEKA environment.

Table 4.2: The summary of training dataset results obtained from the logistic regression classifier model

Metrics	Value
---------	-------

Kappa statistic	0.9143
Mean absolute error	0.063
Root mean squared error	0.1835
Relative absolute error	13.5776%
Root relative squared error	38.1076%
Recall or Sensitivity	0.975
Specificity	0.935
Precision	0.963
ROC Area	0.982
F-Measure	0.969
MCC	0.914
PRC Area	0.984

Time taken to build model: 0.32 seconds

=== Stratified cross-validation ===
 === Summary ===

Correctly Classified Instances	1578	96.0438 %
Incorrectly Classified Instances	65	3.9562 %
Kappa statistic	0.9143	
Mean absolute error	0.063	
Root mean squared error	0.1835	
Relative absolute error	13.5776 %	
Root relative squared error	38.1076 %	
Total Number of Instances	1643	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.935	0.025	0.956	0.935	0.945	0.914	0.982	0.979	HL
	0.975	0.065	0.963	0.975	0.969	0.914	0.982	0.986	LL
Weighted Avg.	0.960	0.050	0.960	0.960	0.960	0.914	0.982	0.984	

=== Confusion Matrix ===

a	b	<-- classified as
561	39	a = HL
26	1017	b = LL

Fig 4.2: The logistic regression classifier model showing the performance of the training dataset

The Multilayer Perceptron (MLP)

The multilayer perceptron classifier model was the next to be built following the logistic regression model. The multilayer perceptron classifier is an artificial neural network with several layers, which includes the input layer, hidden layer and output layer (Kantardzic, 2011). This classifier is the most popular artificial neural network model used for classification tasks in EDM (Mueen et al, 2016).

Table 4.3 shows the summary of the results obtained from the modelling process using the multilayer perceptron algorithm and the diagram in Fig 4.3 illustrates the performance of the model in WEKA environment.

Table 4.3: Summary of training dataset results obtained from the multilayer perceptron classifier model

Metrics	Value
Kappa statistic	0.9381
Mean absolute error	0.0302
Root mean squared error	0.1560
Relative absolute error	6.5135%
Root relative squared error	32.3991%
Recall or Sensitivity	0.983
Specificity	0.952
Precision	0.972
ROC Area	0.992
F-Measure	0.978
MCC	0.938
PRC Area	0.993

Time taken to build model: 51.48 seconds

=== Stratified cross-validation ===
 === Summary ===

Correctly Classified Instances	1596	97.1394 %
Incorrectly Classified Instances	47	2.8606 %
Kappa statistic	0.9381	
Mean absolute error	0.0302	
Root mean squared error	0.156	
Relative absolute error	6.5135 %	
Root relative squared error	32.3991 %	
Total Number of Instances	1643	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.952	0.017	0.969	0.952	0.960	0.938	0.992	0.991	HL
	0.983	0.048	0.972	0.983	0.978	0.938	0.992	0.993	LL
Weighted Avg.	0.971	0.037	0.971	0.971	0.971	0.938	0.992	0.992	

=== Confusion Matrix ===

a	b	<-- classified as
571	29	a = HL
18	1025	b = LL

Fig 4.3: The multilayer perceptron classifier model showing the performance of the training dataset

The Naïve Bayes Classifier

The fourth model built is the naïve Bayes classifier model. The naïve Bayes classifier is a common method used for classification task in EDM, which employs the Bayes theorem for conditional probabilities (Shahiri et al, 2015). It is termed “naïve” because it assumes that all attributes are independent of each other, which means that the probability of one attribute does not affect another (Osmanbegovic & Suljic, 2012).

Table 4.4 shows the summary of the results obtained from the modelling process using the naïve Bayes algorithm and the diagram in Fig 4.4 shows the performance of the model in WEKA environment.

Table 4.4: The summary of training dataset results obtained from the Naïve Bayes classifier model

Metrics	Value
Kappa statistic	0.7601
Mean absolute error	0.1164
Root mean squared error	0.2901
Relative absolute error	25.0931%
Root relative squared error	60.2594%
Recall or Sensitivity	0.919
Specificity	0.838
Precision	0.908
ROC Area	0.945
F-Measure	0.913
MCC	0.760
PRC Area	0.958

```

Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      1461           88.9227 %
Incorrectly Classified Instances    182            11.0773 %
Kappa statistic                    0.7601
Mean absolute error                 0.1164
Root mean squared error            0.2901
Relative absolute error            25.0931 %
Root relative squared error        60.2594 %
Total Number of Instances         1643

=== Detailed Accuracy By Class ===

                TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                -----  -----  -
                0.838   0.081   0.855     0.838   0.847     0.760   0.945   0.939   HL
                0.919   0.162   0.908     0.919   0.913     0.760   0.945   0.958   LL
Weighted Avg.   0.889   0.132   0.889     0.889   0.889     0.760   0.945   0.951

=== Confusion Matrix ===

  a  b  <-- classified as
503 97 |  a = HL
 85 958 |  b = LL

```

Fig 4.4: The Naïve Bayes classifier model showing the performance of the training dataset

The Sequential Minimal Optimization (SMO)

The sequential minimal optimization algorithm is an improvement on the algorithms used in support vector machine (SVM). The SVM is a classification model that searches and obtains the largest margin between two classes of data (Aggarwal, 2015). The SMO algorithm is a fast and simple algorithm that solves the problems with quadratic programming encountered in SVM models by breaking them into manageable problems and solving them analytically (Aruna & Rajagopalan, 2011).

Table 4.5 shows the summary of the results obtained from the modelling process using the sequential minimal optimization algorithm and Fig 4.5 depicts the performance of the model in WEKA environment.

Table 4.5: The summary of training dataset results obtained from the sequential minimal optimization classifier model

Metrics	Value
Kappa statistic	0.9164
Mean absolute error	0.0383
Root mean squared error	0.1958

Relative absolute error	8.2693%
Root relative squared error	40.6697%
Recall or Sensitivity	0.984
Specificity	0.923
Precision	0.957
ROC Area	0.954
F-Measure	0.970
MCC	0.917
PRC Area	0.952

Time taken to build model: 1.19 seconds

=== Stratified cross-validation ===

=== Summary ===

```

Correctly Classified Instances      1580          96.1656 %
Incorrectly Classified Instances    63           3.8344 %
Kappa statistic                    0.9164
Mean absolute error                 0.0383
Root mean squared error             0.1958
Relative absolute error              8.2693 %
Root relative squared error         40.6697 %
Total Number of Instances          1643

```

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.923	0.016	0.970	0.923	0.946	0.917	0.954	0.924	HL
	0.984	0.077	0.957	0.984	0.970	0.917	0.954	0.952	LL
Weighted Avg.	0.962	0.055	0.962	0.962	0.961	0.917	0.954	0.942	

=== Confusion Matrix ===

```

  a    b  <-- classified as
554  46 |   a = HL
 17 1026 |   b = LL

```

Fig 4.5: The sequential minimal optimization classifier model showing the performance of the training dataset

4.2.2 Presentation and Interpretation of Test Dataset

After using the training dataset to build models with the five classifiers, which are the J48, LR, MLP, NV and SMO, the study used each model built for testing with the test dataset set aside to discover how well the model generalises. A model that performs well with a training dataset and performs badly with the test dataset is a biased model and considered not suitable for use with real world data (Lever et al, 2016). The presentation of results and discussions for the five algorithms used on test dataset are presented next

The J48 Decision Tree

The J48 decision tree model built in Section 4.2.1 is used for testing the 30% dataset set aside for the testing purpose. The J48 decision tree is used to test with the model built with the J48 decision tree algorithm and the results from the testing is presented in Table 4.6 and Fig 4.6.

Table 4.6 summarises the results obtained from the modelling process using the J48 decision tree algorithm and the diagram in Fig 4.6 gives the performance of the model in WEKA environment.

Table 4.6: The summary of test dataset results obtained from the J48 classifier model

Metrics	Value
Kappa statistic	0.9349
Mean absolute error	0.0443
Root mean squared error	0.1685
Recall or Sensitivity	0.991
Specificity	0.934
Precision	0.963
ROC Area	0.976
F-Measure	0.977
MCC	0.936
PRC Area	0.977

```

=== Re-evaluation on test set ===

User supplied test set
Relation:      NewDataset2-weka.filters.unsupervised.instance.Resample-S1-Z70.0-no-replacement-V
Instances:    unknown (yet). Reading incrementally
Attributes:   25

=== Summary ===

Correctly Classified Instances      684          97.0213 %
Incorrectly Classified Instances    21           2.9787 %
Kappa statistic                    0.9349
Mean absolute error                 0.0443
Root mean squared error             0.1685
Total Number of Instances          705

=== Detailed Accuracy By Class ===

          TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
          0.934   0.009   0.984     0.934   0.958     0.936   0.976    0.967    HL
          0.991   0.066   0.963     0.991   0.977     0.936   0.976    0.977    LL
Weighted Avg.   0.970   0.046   0.971     0.970   0.970     0.936   0.976    0.973

=== Confusion Matrix ===

  a  b  <-- classified as
239 17 |  a = HL
  4 445|  b = LL

```

Fig 4.6: The J48 classifier model showing the performance of the test dataset

The Logistic Regression Classifier

From the design of the logistic regression classifier model built and shown in Section 4.2.1, the dataset set aside for testing establishes how well the trained model can generalise. The logistic regression classifier tests the model built with the logistic regression algorithm and the results are presented in Table 4.7 and Fig 4.7.

Table 4.7 shows the summary of the results obtained from the modelling process using the logistic regression algorithm and the diagram in Fig 4.7 depicts the performance of the model in WEKA environment.

Table 4.7: Summary of test dataset results obtained from the Logistic Regression classifier model

Metrics	Value
Kappa statistic	0.9227
Mean absolute error	0.0608
Root mean squared error	0.1724
Recall or Sensitivity	0.982
Specificity	0.934
Precision	0.963
ROC Area	0.988
F-Measure	0.972
MCC	0.923
PRC Area	0.992

```

=== Re-evaluation on test set ===

User supplied test set
Relation:      NewDataset2-weka.filters.unsupervised.instance.Resample-S1-270.0-no-replacement-V
Instances:     unknown (yet). Reading incrementally
Attributes:    25

=== Summary ===

Correctly Classified Instances      680           96.4539 %
Incorrectly Classified Instances     25           3.5461 %
Kappa statistic                     0.9227
Mean absolute error                  0.0608
Root mean squared error              0.1724
Total Number of Instances           705

=== Detailed Accuracy By Class ===

                TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                0.934   0.018   0.968     0.934   0.950     0.923   0.988    0.986    HL
                0.982   0.066   0.963     0.982   0.972     0.923   0.988    0.992    LL
Weighted Avg.   0.965   0.049   0.965     0.965   0.964     0.923   0.988    0.990

=== Confusion Matrix ===

  a  b  <-- classified as
239 17 | a = HL
  8 441| b = LL

```

Fig 4.7: The Logistic Regression classifier model showing the performance of the test dataset

The Multilayer Perceptron (MLP)

The multilayer perceptron algorithm tests the model designed with the multilayer perceptron classifier in Section 4.2.1 with the 30% dataset set aside for the testing. The results from the testing are presented below in Table 4.8 and Fig 4.8.

Table 4.8 shows the summary of the results obtained from the modelling process using the multilayer perceptron algorithm and the diagram in Fig 4.8 depicts the performance of the model in WEKA environment.

Table 4.8: The summary of test dataset results obtained from the Multilayer Perceptron classifier model

Metrics	Value
Kappa statistic	0.9631

Mean absolute error	0.0195
Root mean squared error	0.1205
Recall or Sensitivity	0.991
Specificity	0.969
Precision	0.982
ROC Area	0.998
F-Measure	0.987
MCC	0.963
PRC Area	0.999

=== Re-evaluation on test set ===

User supplied test set

Relation: NewDataset2-weka.filters.unsupervised.instance.Resample-S1-Z70.0-no-replacement-V
Instances: unknown (yet). Reading incrementally
Attributes: 25

=== Summary ===

Correctly Classified Instances	693	98.2979 %
Incorrectly Classified Instances	12	1.7021 %
Kappa statistic	0.9631	
Mean absolute error	0.0195	
Root mean squared error	0.1205	
Total Number of Instances	705	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.969	0.009	0.984	0.969	0.976	0.963	0.998	0.997	HL
	0.991	0.031	0.982	0.991	0.987	0.963	0.998	0.999	LL
Weighted Avg.	0.983	0.023	0.983	0.983	0.983	0.963	0.998	0.998	

=== Confusion Matrix ===

a	b	<-- classified as
248	8	a = HL
4	445	b = LL

Fig 4.8: The Multilayer Perceptron classifier model showing the performance of the test dataset

The Naïve Bayes Classifier

With the naïve Bayes classifier model built and presented in Section 4.2.1; the test dataset set aside confirms how well the trained model generalises. The naïve Bayes classifier tests the model built with the naïve Bayes algorithm and the results from the testing are presented in Table 4.9 and Fig 4.9.

Table 4.9 summarises the results obtained from the modelling process using the naïve Bayes algorithm and the diagram in Fig 4.9 depicts the performance of the model in WEKA environment.

Table 4.9: The summary of test dataset results obtained from the Naïve Bayes classifier model

Metrics	Value
Kappa statistic	0.7673
Mean absolute error	0.1168
Root mean squared error	0.2918
Recall or Sensitivity	0.913
Specificity	0.855
Precision	0.917
ROC Area	0.944
F-Measure	0.915
MCC	0.767
PRC Area	0.956

```

=== Re-evaluation on test set ===

User supplied test set
Relation:      NewDataset2-weka.filters.unsupervised.instance.Resample-S1-270.0-no-replacement-V
Instances:     unknown (yet). Reading incrementally
Attributes:    25

=== Summary ===

Correctly Classified Instances      629           89.2199 %
Incorrectly Classified Instances    76           10.7801 %
Kappa statistic                    0.7673
Mean absolute error                 0.1168
Root mean squared error             0.2918
Total Number of Instances          705

=== Detailed Accuracy By Class ===

          TP Rate  FP Rate  Precision  Recall   F-Measure  MCC     ROC Area  PRC Area  Class
          0.855   0.087   0.849     0.855   0.852     0.767   0.944    0.938    HL
          0.913   0.145   0.917     0.913   0.915     0.767   0.944    0.956    LL
Weighted Avg.   0.892   0.124   0.892     0.892   0.892     0.767   0.944    0.950

=== Confusion Matrix ===

  a  b  <-- classified as
219 37 |  a = HL
 39 410 |  b = LL

```

Fig 4.9: The Naïve Bayes classifier model showing the performance of the test dataset

The Sequential Minimal Optimization (SMO)

The training dataset model designed with the sequential minimal optimization algorithm is presented in Section 4.2.1 and with this model, the dataset set aside for testing was used to test and establish how well the trained data generalises. The SMO classifier is used to test the model built with the SMO algorithm and the results from the testing is presented in Table 4.10 and Fig 4.10.

Table 4.10 shows the summary of the results obtained from the modelling process using the sequential minimal optimization algorithm and the diagram in Fig 4.10 depicts the performance of the model in WEKA environment.

Table 4.10: Summary of test dataset results obtained from the Sequential Minimal Optimization classifier model

Type of Error	Value
Kappa statistic	0.9256
Mean absolute error	0.0340
Root mean squared error	0.1845
Recall or Sensitivity	0.987
Specificity	0.930
Precision	0.961
ROC Area	0.958
F-Measure	0.974
MCC	0.926
PRC Area	0.957

```

=== Re-evaluation on test set ===

User supplied test set
Relation:      NewDataset2-weka.filters.unsupervised.instance.Resample-S1-Z70.0-no-replacement-V
Instances:    unknown (yet). Reading incrementally
Attributes:   25

=== Summary ===

Correctly Classified Instances      681          96.5957 %
Incorrectly Classified Instances    24           3.4043 %
Kappa statistic                    0.9256
Mean absolute error                 0.034
Root mean squared error             0.1845
Total Number of Instances          705

=== Detailed Accuracy By Class ===

          TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
          0.930   0.013   0.975     0.930   0.952     0.926   0.958    0.932    HL
          0.987   0.070   0.961     0.987   0.974     0.926   0.958    0.957    LL
Weighted Avg.   0.966   0.050   0.966     0.966   0.966     0.926   0.958    0.948

=== Confusion Matrix ===

  a  b  <-- classified as
238 18 |  a = HL
  6 443 |  b = LL

```

Fig 4.10: The Sequential Minimal Optimization classifier model showing the performance of the test dataset

4.2.3 Performance of Classifiers and Findings

To determine the best classifier model for the type of data used in this study, five classification algorithms were used for modelling the data and their performances compared. This section deals with discussion on the performance and findings from the comparisons of the five algorithms used. Table 4.11 compares the performance of the classifier models based on correctly and incorrectly classified student data for the training dataset. From the table, MLP algorithm, correctly classified the highest number students (1596) for the entire training dataset and misclassified the lowest number of students (47). SMO has the next best performance classifier with 1580 records classified correctly and 63 misclassifications. The logistic regression follows SMO with the classification margin of two less than SMO, 1578 records are correctly classified and 65 records are incorrectly classified. J48 correctly classified 1566 student records and misclassified 77 records. For this study, the naïve Bayes shows the least performance with 182 misclassified records and 1461 correctly classified records.

For the performance of the algorithms in correctly classifying students in HL class, MLP outperformed the other algorithms; while for the performance of the algorithms in correctly classifying students in LL class, SMO performs best. However, the difference in performance between SMO and MLP in correctly classifying students in LL class is just one record.

The focus in this study is to classify low performing students into two groups of HL and LL with the aim of providing students in the LL group with the urgent intervention assistance they need to perform better. The performance of the classifier models shows the SMO as the best classifier for correctly identifying students in LL class; however, the MLP is the best classifier in correctly classifying all student records with a difference of one in correctly classifying students in LL group. This study considers the MLP classifier as the most suitable classifier for the data used in this study for the training dataset; however, the study looks at performance of the classifiers with the test dataset before offering binding conclusions.

Table 4.11: Comparison of the classifier models performance based on correctly and incorrectly classified student data for the training dataset

	J48	LR	MLP	NB	SMO
Correctly classified students	1566	1578	1596	1461	1580
Incorrectly classified students	77	65	47	182	63
Correctly classified HL students	547	561	571	503	554
Incorrectly classified HL students	53	39	29	97	46

Correctly classified LL students	1019	1017	1025	958	1026
Incorrectly classified LL students	24	26	18	85	17

After looking at the performance of the five algorithms on the training dataset, the study compared the performance of the five algorithms on the test dataset. The table in Table 4.12 compares the performance of the classifier models based on correctly and incorrectly classified student data for the test dataset. From the table, MLP algorithm, correctly classified the highest number students (693) for the entire test dataset and misclassified the lowest number of students (12). J48 exhibited the next best performance with 684 records classified correctly and 21 misclassifications. SMO follows J48 with 681 records correctly classified and 24 incorrectly classified records. Logistic regression is next with 680 correctly classified student records and 25 misclassified records. For the test dataset, the naïve Bayes still shows the least performance with 76 misclassified records and 629 correctly classified records.

For the performance of the algorithms in correctly classifying students in HL class, MLP outperformed the other algorithms. For the performance of the algorithms in correctly classifying students in LL class, MLP and J48 achieved the same level of performance.

From the discussion above, the MLP classifier shows the best performance in correctly classifying all student records, students in HL and LL classes; therefore, this study considers the MLP classifier as the most suitable classifier for the data used based on its performance for both the training and test dataset.

For the purpose of more analysis on the choice of the best classifier for the data used in this study, the study further reviewed the performance of the five algorithms based on the six selected evaluation metrics mentioned earlier. Table 4.13 and Table 4.14 show the comparison of the classifiers performance using the six selected metrics on the training and test dataset respectively.

Table 4.12: Comparison of the classifier models performance based on correctly and incorrectly classified student data for the test dataset

	J48	LR	MLP	NB	SMO
Correctly classified students	684	680	693	629	681
Incorrectly classified students	21	25	12	76	24
Correctly classified HL students	239	239	248	219	238
Incorrectly classified HL students	17	17	8	37	18
Correctly classified LL students	445	441	445	410	443
Incorrectly classified LL students	4	8	4	39	6

The six selected metrics used to evaluate the performance of the classifier models built in this research are the values of recall or sensitivity, specificity, ROC area, F-Measure Kappa statistics and root mean squared error (RMSE). The first metric considered is the recall or sensitivity value. The recall value measures the proportion of correctly identified LL records against all the LL records. For the training dataset shown in Table 4.13, the results show that all the classifiers achieve a recall value of over 90%. This indicates that all the classifiers are very sensitive, suggesting that they perform very well in classifying students in LL group. However, SMO achieved the highest recall value of 98.4% followed closely by the MLP classifier with 98.3% and the naïve Bayes classifier has the lowest recall value of 91.9%.

Table 4.13: Comparison of the classifiers performance on the training dataset using the six selected metrics

Model	Recall	Specificity	ROC	F-Measure	Kappa	RMSE
J48	0.977	0.912	0.961	0.964	0.8979	0.2085
LR	0.975	0.935	0.982	0.969	0.9143	0.1835
MLP	0.983	0.952	0.992	0.978	0.9381	0.1560
NV	0.919	0.838	0.945	0.913	0.7601	0.2901
SMO	0.984	0.923	0.954	0.970	0.9164	0.1958

The specificity value measures the rate of correctly classified HL records to the entire HL records. For the training dataset, all the classifiers achieved over 80% specificity values. This performance indicates that all the classifiers are very specific and they are capable of classifying students in HL group. The MLP classifier attained the highest specificity value of 95.2% and the naïve Bayes classifier has the lowest specificity value of 83.8%. The ROC area is a reliable measure of classifier performance that plots sensitivity against specificity; a ROC area value close to 100% indicates the model's ability to classify students correctly in the group they belong. Results from Table 4.13 for the training dataset show that all the classifiers have good ROC area values of over 90%; however, the MLP classifier model has the highest ROC value of 92.2% and the naïve Bayes has the lowest ROC area value of 94.5%.

The F-Measure metric determines the average of precision and recall considering only high-risk students; the F-Measure can separately determine the performance of different classes. Table 4.13 shows the results for high-risk students as the study is interested in this set of students. The MLP classifier has the highest value of 97.8% and the naïve Bayes model has the lowest value of 91.8%. The next metric is the Kappa statistic metric; a value close to one indicates that both classes concur

on the classification of students as either high risk or low risk. The MLP has the highest Kappa value of 0.9381 and the classifier with the lowest kappa value of 0.7601 is the naïve Bayes classifier. The kappa values for all the five classifiers are suitable for use as they have above 0.60, which is the benchmark value.

The root mean square error (RMSE) is the final metric considered in this research. RMSE is the average error between the values predicted and the actual values. When a classifier has a low RMSE value, it shows that the classifier performs well. The results from the model demonstrate that the MLP classifier has the lowest RMSE value of 0.1560 and the naïve Bayes classifier has the highest RMSE value of 0.2901. Subsequently, the study looks at the performance of the classifiers on the test dataset based on the six selected metrics of evaluation.

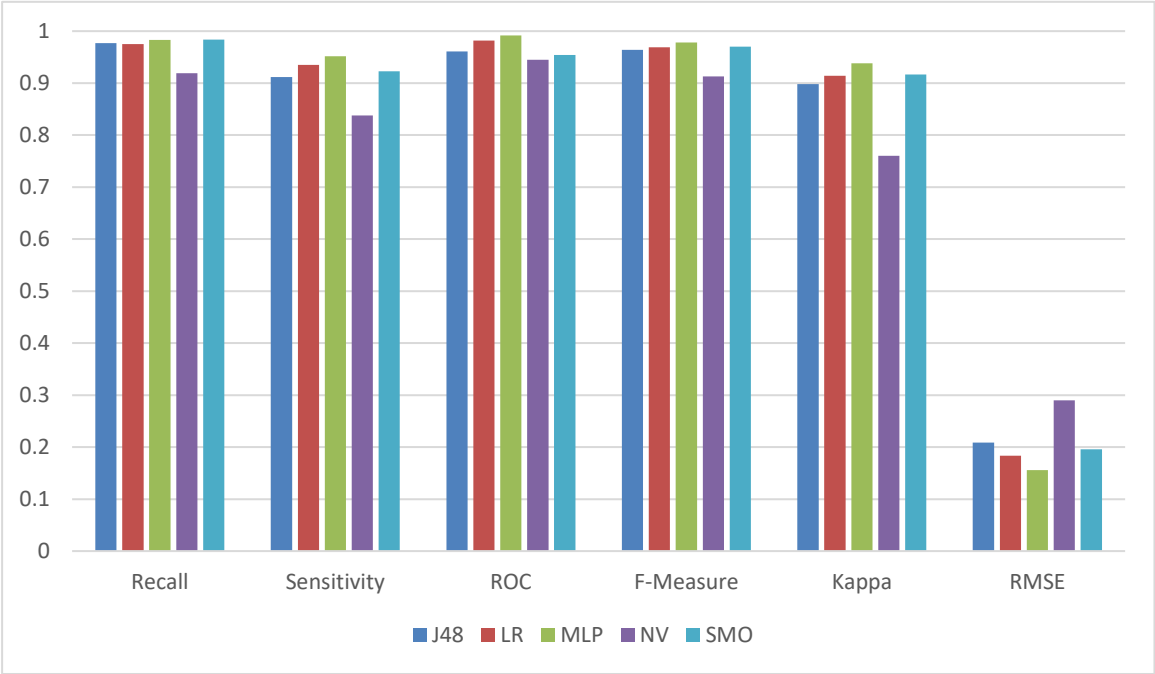


Fig 4.11: Summary of the classifiers performance on the training dataset using the six selected metrics

The Figure 4.11 represents the summary of the performance of the classifiers on the training dataset using the six selected metrics in the form of a bar chart. The chart shows that the MLP model represented by the green bar is best in all metrics of evaluation except for recall value where the SMO model is better by a small margin. The naïve Bayes model with purple bar shows the worst performance for the training dataset in all the six selected metrics of evaluation.

Table 4.14: Comparison of the classifiers performance on the test dataset using the six selected metrics

Model	Recall	Specificity	ROC	F-Measure	Kappa	RMSE
J48	0.991	0.934	0.976	0.977	0.9349	0.1685
LR	0.982	0.934	0.988	0.972	0.9227	0.1724
MLP	0.991	0.969	0.998	0.987	0.9631	0.1205
NV	0.913	0.855	0.944	0.915	0.7673	0.2918
SMO	0.987	0.930	0.958	0.974	0.9256	0.1845

For the test dataset result available in Table 4.14, the result shows that all the classifiers achieve a recall value of over 90%. This indicates that all the classifiers are very sensitive, suggesting that they perform very well in classifying students in LL group. However, the MLP and J48 classifiers performed best with values of 99.1% and the naïve Bayes classifier has the lowest recall value of 91.3%. The specificity value measures the rate of correctly classified HL records to the entire HL records. For the test dataset, all the classifiers achieved over 80% specificity values, indicating that all the classifiers are very specific. The conclusion is that they perform very well in classifying students in HL group. The MLP classifier attained the highest specificity value of 96.9% and the naïve Bayes classifier has the lowest specificity value of 85.5%. The ROC area is a reliable measure of classifier performance that plots sensitivity against specificity. A ROC area value close to 100% indicates the model's ability to classify students correctly in the group they belong. Results from Table 4.14 for the test dataset shows that all the classifiers have good ROC area values of over 90%; however, the MLP classifier model has the highest ROC value of 99.8% and the naïve Bayes has the lowest ROC area value of 94.4%.

The F-Measure metric determines the average of precision and recall considering only high-risk students. The F-Measure can separately determine the performance of different classes. The Table 4.13 shows the results for high-risk students as the study is interested in these set of students. The MLP classifier has the highest value of 98.7% and the naïve Bayes model has the lowest value of 91.5%. The next metric is the Kappa statistic metric; a value close to one indicates that both classes concur on the classification of students as either high risk of low risk. The MLP has the highest Kappa value of 0.9631 and the classifier with the lowest kappa value of 0.7673 is the naïve Bayes classifier. The kappa values for all the five classifiers are suitable for use as they have above 0.60, which is the benchmark value.

The root mean square error (RMSE) is the final metric considered in this research. RMSE is the average error between the values predicted and the actual values. When a classifier has a low

RMSE value, it shows that the classifier performs well. The results from the model show that the MLP classifier has the lowest RMSE value of 0.1205 and the naïve Bayes classifier has the highest RMSE value of 0.2918.

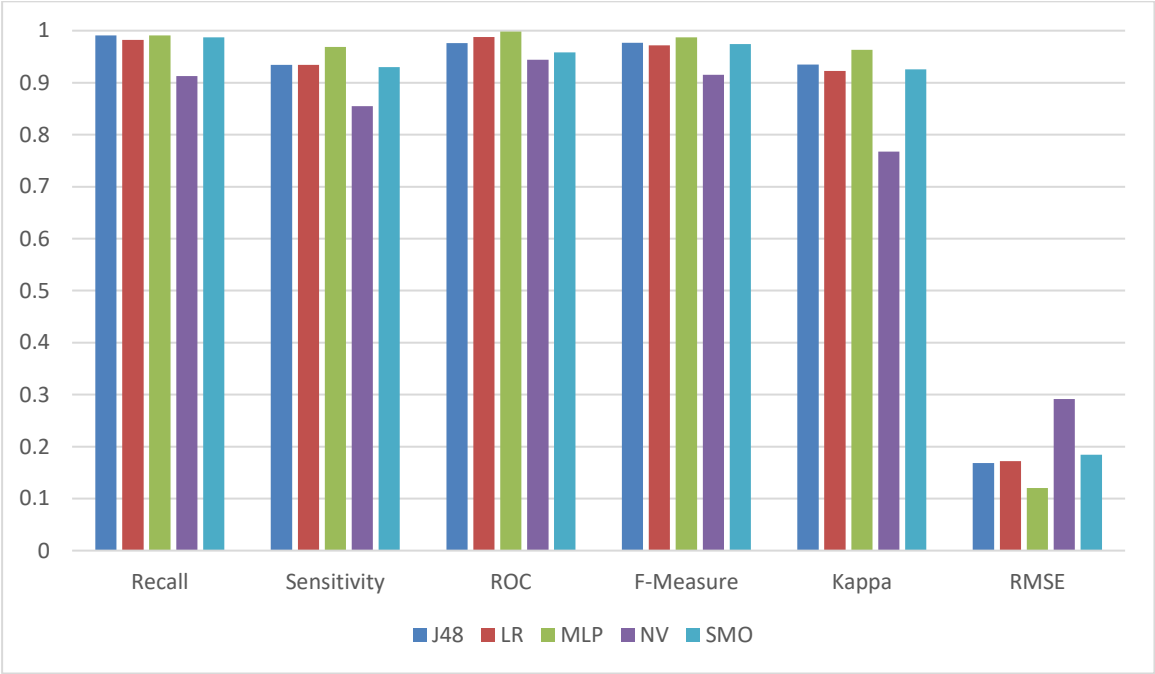


Fig 4.12: Summary of the classifiers performance on the test dataset using the six selected metrics

The result of the classifiers used on the test dataset shows some improvement compared to the training dataset based on the six selected metrics used for evaluating the models. The differences in performances are relatively low for models with better performance as is the case of the naïve Bayes classifier. The improvements in the model performance for the test dataset indicates that model built is highly unbiased and can generalise well for real world data.

Figure 4.12 represents the summary of the classifiers performance on the test dataset using the six selected metrics on a bar chart. The chart shows that the MLP model represented by the green bar is the best in all metrics of evaluation and the naïve Bayes model with purple bar shows the worst performance for the test dataset in all the six selected metrics of evaluation.

4.3 Presentation of Feature Selection

Feature selection in data mining can assist models by finding the most relevant features in the dataset, reducing the model’s complexities and improving the accuracy of the model (Neumann et al, 2016). In selecting optimal features, the decision to use either complete dataset or the training

dataset is necessary. However, making use of the training dataset offers the design of good model performance with reduced features, which is an appealing aspect in machine learning. Therefore, this study used the training dataset to identify and confirm the most relevant attributes out of the training dataset that contains 1643 student records.

This section presents the results of feature selection using the WEKA modelling tool. This study used the training dataset and algorithms for feature selection available on WEKA called ‘attribute selector’. The algorithms rank the features in order of importance from best to least. The four attribute selectors used to accomplish the feature selection task are Correlation, Gain ratio, Information gain and ReliefF. With the results acquired from the ranking of these algorithms, this study implemented a method of determining the best features by consecutively modelling starting with the top four set of attributes ranked until all the 24 attributes were modelled. This consecutive modelling pattern is adopted from the research carried out by Ramaswani and Bhaskaran (2009) where they employed different attribute selectors to determine the best features for predicting students’ grades.

Correlation

The diagram in Fig 4.13 shows the correlation algorithm in WEKA ranking the features in order of importance from highest to lowest. From the diagram, the third column representing the attribute name shows that the four most important attributes are *Sponsor qualification*, *Secondary school type*, *Work and Study* and *University accommodation* while the least four features for this algorithm are *Family size*, *Smart phone assistance*, *Attended primary school* and *Marital status*.

```

=== Attribute selection 10 fold cross-validation (stratified), seed: 1 ===
average merit      average rank  attribute
0.51 +- 0.007      1 +- 0       8 SponQual
0.426 +- 0.005     2.4 +- 0.49  5 SecType
0.424 +- 0.009     2.6 +- 0.49  12 WorkStudy
0.381 +- 0.007     4.4 +- 0.49  13 UniAcc
0.382 +- 0.007     4.6 +- 0.49  6 SecArea
0.321 +- 0.006     6 +- 0       21 SptAc
0.293 +- 0.006     7.4 +- 0.49  14 BeAdmYrs
0.293 +- 0.006     7.6 +- 0.49  7 SponType
0.273 +- 0.004     9.6 +- 0.49  10 SponSup
0.27 +- 0.008      9.8 +- 0.87  15 JambCou
0.261 +- 0.009    10.8 +- 0.87  2 Age
0.252 +- 0.004    11.8 +- 0.4  17 WkStud
0.222 +- 0.004    13.4 +- 0.49  24 AveSc
0.219 +- 0.007    13.7 +- 0.64  9 SponInc
0.203 +- 0.005    15.3 +- 0.46  22 JambSc
0.199 +- 0.008    15.7 +- 0.78  19 SmPhn
0.188 +- 0.008    16.9 +- 0.3  18 PgDeg
0.162 +- 0.004    18 +- 0       23 PumeSc
0.151 +- 0.004    19 +- 0       1 Sex
0.125 +- 0.002    20.1 +- 0.3  16 CouInt
0.119 +- 0.004    20.9 +- 0.3  11 FamSize
0.088 +- 0.004    22 +- 0       20 SmPhnAss
0.045 +- 0.007    23 +- 0       4 AtPri
0.01 +- 0.007     24 +- 0       3 MarStat

```

Fig 4.13: Correlation ranked features from the most important to the least important

Gain Ratio

The feature ranked with Gain Ratio is presented in Fig 4.14. From the third column, the best four attributes are *Sponsor qualification*, *Work and study*, *Secondary school type* and *Average SSCE score* while the least four features are *Sex*, *Family size*, *Attended primary school* and *Marital status*. The Gain Ratio algorithm shares three best attributes (*Sponsor qualification*, *Work and study*, *Secondary school type*) and three least attributes (*Family size*, *Attended primary school*, *Marital status*) with the Correlation algorithm.

=== Attribute selection 10 fold cross-validation (stratified), seed: 1 ===

average merit	average rank	attribute
0.194 +- 0.006	1 +- 0	8 SponQual
0.142 +- 0.006	2.4 +- 0.49	12 WorkStudy
0.136 +- 0.003	3.2 +- 0.98	5 SecType
0.132 +- 0.002	3.5 +- 0.67	24 AveSc
0.126 +- 0.003	5 +- 0.45	17 WkStud
0.119 +- 0.005	6 +- 0.45	13 UniAcc
0.109 +- 0.004	6.9 +- 0.3	6 SecArea
0.097 +- 0.003	8.2 +- 0.4	7 SponType
0.092 +- 0.003	8.9 +- 0.54	22 JambSc
0.085 +- 0.003	9.9 +- 0.3	10 SponSup
0.077 +- 0.003	11.2 +- 0.4	21 SptAc
0.075 +- 0.003	11.9 +- 0.54	9 SponInc
0.069 +- 0.004	12.9 +- 0.3	18 PgDeg
0.061 +- 0.002	14.2 +- 0.4	14 BeAdmYrs
0.054 +- 0.004	15.4 +- 0.66	2 Age
0.053 +- 0.003	15.5 +- 0.81	15 JambCou
0.047 +- 0.004	17.4 +- 0.8	19 SmPhn
0.044 +- 0.002	18 +- 0.63	23 PumeSc
0.042 +- 0.002	18.5 +- 0.81	16 CouInt
0.019 +- 0.002	20.1 +- 0.3	20 SmPhnAss
0.016 +- 0.001	20.9 +- 0.3	1 Sex
0.012 +- 0.001	22.1 +- 0.3	11 FamSize
0.009 +- 0.003	22.9 +- 0.3	4 AtPri
0 +- 0.001	24 +- 0	3 MarStat

Fig 4.14: Gain Ratio ranked features from the most important to the least important

Information Gain

The diagram (Fig 4.15) presents the features ranked by the Information Gain algorithm. The attributes considered as the best four with this algorithm are *Sponsor qualification*, *Weekly study time*, *Average SSCE score* and *Sponsor type* while the least four features are, *Family size*, *Sex*, *Attended primary school* and *Marital status*. This algorithm shares the *Sponsor qualification* attribute as the best attribute with both Correlation and Gain Ratio algorithms. The *Average SSCE score* attribute is also part of the top four attributes for the Gain Ratio algorithm. The least features for the Information Gain algorithm are the same as the Gain Ratio algorithm, thereby sharing the three least attributes (*Family size*, *Attended primary school*, *Marital status*) with the Correlation algorithm.

=== Attribute selection 10 fold cross-validation (stratified), seed: 1 ===

average merit	average rank	attribute
0.223 +- 0.006	1 +- 0	8 SponQual
0.194 +- 0.005	2 +- 0	17 WkStud
0.18 +- 0.004	3 +- 0	24 AveSc
0.145 +- 0.004	4.1 +- 0.3	7 SponType
0.135 +- 0.005	5.8 +- 1.17	22 JambSc
0.135 +- 0.005	5.8 +- 0.75	10 SponSup
0.132 +- 0.003	6.8 +- 0.98	5 SecType
0.129 +- 0.006	7.5 +- 0.67	12 WorkStudy
0.113 +- 0.004	9.1 +- 0.3	9 SponInc
0.106 +- 0.004	10.3 +- 0.46	6 SecArea
0.103 +- 0.004	10.8 +- 0.87	13 UniAcc
0.092 +- 0.006	11.8 +- 0.4	18 PgDeg
0.078 +- 0.003	13.3 +- 0.46	14 BeAdmYrs
0.077 +- 0.003	13.7 +- 0.46	21 SptAc
0.067 +- 0.003	15.2 +- 0.4	23 PumeSc
0.062 +- 0.003	15.8 +- 0.4	16 CouInt
0.053 +- 0.003	17.2 +- 0.4	15 JambCou
0.048 +- 0.003	17.8 +- 0.4	2 Age
0.028 +- 0.002	19 +- 0	20 SmPhnAss
0.027 +- 0.002	20 +- 0	19 SmPhn
0.019 +- 0.001	21.1 +- 0.3	11 FamSize
0.016 +- 0.001	21.9 +- 0.3	1 Sex
0.002 +- 0	23 +- 0	4 AtPri
0 +- 0	24 +- 0	3 MarStat

Fig 4.15: Information Gain ranked features from the most important to the least important

ReliefF

The ranked attributes using the ReliefF algorithm is presented in Fig 4.16. From the third column the attributes considered as the best four for this algorithm are *Family size*, *Sponsor type*, *Weekly study time* and *Sponsor qualification* while the least four attributes are *Course from Jamb*, *Own smart phone*, *Marital status* and *Attended primary school*. All the algorithms share the *Sponsor qualification* attribute as part of the top four. The ReliefF algorithm shares the *Sponsor type* and *Weekly study time* attributes as top four attributes with the Information Gain algorithm. The attributes shared between all the four algorithms as part of the least four attributes are the *Attended primary school* and *Marital status* attributes.

=== Attribute selection 10 fold cross-validation (stratified), seed: 1 ===

average merit	average rank	attribute
0.375 +- 0.009	1 +- 0	11 FamSize
0.353 +- 0.008	2 +- 0	7 SponType
0.309 +- 0.008	3.6 +- 0.66	17 WkStud
0.3 +- 0.01	4.1 +- 1.04	8 SponQual
0.3 +- 0.005	4.4 +- 0.66	10 SponSup
0.283 +- 0.009	6.5 +- 0.92	2 Age
0.278 +- 0.006	6.9 +- 0.7	14 BeAdmYrs
0.274 +- 0.007	7.5 +- 0.67	22 JambSc
0.26 +- 0.006	9.2 +- 0.6	9 SponInc
0.251 +- 0.008	10.7 +- 1.1	24 AveSc
0.246 +- 0.006	11.1 +- 0.94	23 PumeSc
0.241 +- 0.008	11.8 +- 1.08	16 CouInt
0.234 +- 0.007	12.9 +- 1.45	18 PgDeg
0.228 +- 0.008	14.2 +- 1.54	21 SptAc
0.219 +- 0.009	15.8 +- 1.33	20 SmPhnAss
0.218 +- 0.01	15.8 +- 1.25	5 SecType
0.219 +- 0.007	15.9 +- 1.3	6 SecArea
0.199 +- 0.007	17.9 +- 0.94	13 UniAcc
0.193 +- 0.006	18.8 +- 0.6	1 Sex
0.181 +- 0.005	20.1 +- 0.54	12 WorkStudy
0.168 +- 0.009	20.8 +- 0.4	15 JambCou
0.075 +- 0.003	22 +- 0	19 SmPhn
0.057 +- 0.006	23 +- 0	3 MarStat
0.017 +- 0.003	24 +- 0	4 AtPri

Fig 4.16: ReliefF ranked features from the most important to the least important

In summary, of all the feature selection algorithms used in this study, the attributes shared as part of the top four attributes by two or more of the algorithms are *Sponsor qualification*, *Sponsor type*, *Secondary school type*, *Work and study*, *Weekly study time* and *Average SSCE score*. This indicates that these attributes could contribute highly to how students are ultimately classified into performance groups. The attributes shared as part of the least four attributes by two or more of the algorithms are *Family size*, *Sex*, *Attended primary school* and *Marital status*, suggesting and closely intimating that these features may contribute least to students' classified group.

To assess the feature selection analysis further, the next section looks at the performance of the algorithms by successively modelling each algorithm from the best four features to the least significant feature.

4.3.1 Performance Evaluation for Selected Features

In evaluating the performance of the algorithms for feature selection, this study carried out consecutive modelling of the attributes ranked for each algorithm. The process followed to achieve

this is by selecting the top four attributes of each algorithm and consecutively adding the next ranked attribute until all the 24 attributes are complete. To evaluate the performance for the selected attributes, the study took note of the ROC and RMSE (abbreviated to RE in the tables) metric values for each set of attributes; these metrics are two widely used measures in evaluating model performance (Caruana & Niculescu-Mizil, 2004).

Correlation

The Table 4.15 presents the results of the consecutive modelling using the Correlation algorithm. From the table, the MLP model achieved the highest ROC value of 0.993 at 15 features and the lowest RMSE value of 0.1410 at 21 features. The logistic regression classifier model achieved the next best ROC value of 0.982 at 21 features and the SMO classifier achieved the next lowest RMSE value of 0.1796 at 20 features. The naïve Bayes classifier has the least performance with ROC value of 0.954 and RMSE value of 0.2700 at 13 features. Since the aim of feature selection is to identify the minimum number of features that attain the best performance for a model; comparing the best feature performance attained between the range of 15-21 features and the overall feature of 24 shows that this algorithm does not achieve the goal of feature selection. Therefore, the study considers the Correlation algorithm as unsuitable for achieving the purpose of feature selection.

Table 4.15: Performance of the five classifiers on Correlation ranked attributes

#F	J48		LR		MLP		NB		SMO	
	ROC	RE	ROC	RE	ROC	RE	ROC	RE	ROC	RE
4	.899	.3235	.893	.3499	.908	.3303	.888	.3538	.785	.4187
5	.913	.2963	.914	.3319	.941	.2928	.910	.3372	.813	.4179
6	.924	.2718	.916	.3317	.955	.2647	.911	.3488	.813	.4179
7	.952	.2423	.927	.3118	.963	.2296	.922	.3262	.849	.3701
8	.955	.2267	.930	.3106	.979	.2065	.917	.3410	.835	.3830
9	.967	.2055	.937	.3038	.978	.1859	.917	.3446	.862	.3515
10	.969	.2058	.943	.2945	.974	.1782	.923	.3367	.859	.3558
11	.964	.1987	.960	.2535	.978	.1838	.936	.3130	.904	.2909
12	.972	.1909	.964	.2422	.988	.1628	.945	.2949	.912	.2736
13	.966	.1874	.971	.2197	.982	.1671	.954	.2700	.924	.2479
14	.967	.1818	.972	.2149	.988	.1532	.949	.2836	.934	.2275
15	.964	.1897	.979	.1964	.993	.1460	.950	.2754	.952	.2019
16	.966	.1903	.979	.1968	.991	.1502	.949	.2763	.949	.2064
17	.967	.1907	.979	.1963	.989	.1471	.946	.2962	.942	.2193
18	.965	.1953	.978	.1953	.988	.1503	.946	.2939	.939	.2288
19	.967	.1922	.978	.1937	.992	.1443	.946	.2906	.942	.2234
20	.968	.1930	.979	.1871	.992	.1414	.947	.2881	.961	.1796

21	.967	.1965	.982	.1835	.989	.1410	.946	.2887	.955	.1911
22	.966	.2019	.982	.1824	.992	.1531	.945	.2906	.954	.1927
23	.966	.2019	.982	.1833	.989	.1505	.946	.2899	.955	.1927
24	.961	.2085	.982	.1835	.992	.1560	.945	.2901	.954	.1958
Bst	.972	.1818	.982	.1824	.993	.1410	.954	.2700	.961	.1796

Gain Ratio

The results of the consecutive modelling using the Gain Ratio algorithm is presented in Table 4.16. From the table, the MLP model achieved the highest ROC value of 0.994 at 14 features and the lowest RMSE value of 0.1471 at 17 features. The logistic regression classifier model achieves achieved the next best ROC value of 0.982 and lowest RMSE value of 0.1824 at 22 features. The naïve Bayes has the least performance with ROC value of 0.947 and RMSE value of 0.2827 at 9 features. The range of optimal features for this algorithm is between 14-22 features and this is also considered as unsuitable in terms of achieving the purpose of feature selection. This study also rules out the Gain Ratio algorithm as unsuitable for achieving the purpose of feature selection.

Table 4.16: Performance of the five classifiers on Gain Ratio ranked attributes

#F	J48		LR		MLP		NB		SMO	
	ROC	RE	ROC	RE	ROC	RE	ROC	RE	ROC	RE
4	.892	.3071	.893	.3461	.920	.2998	.885	.3580	.817	.4143
5	.944	.2654	.936	.2895	.958	.2492	.934	.2899	.874	.3264
6	.948	.2516	.938	.2872	.956	.2387	.937	.2866	.869	.3310
7	.957	.2446	.947	.2813	.963	.2316	.941	.2828	.883	.3150
8	.962	.2262	.950	.2797	.974	.2200	.943	.2925	.894	.3032
9	.970	.2066	.964	.2515	.982	.1948	.947	.2827	.897	.2940
10	.972	.1994	.966	.2510	.980	.1829	.941	.2897	.893	.3022
11	.976	.1895	.966	.2514	.986	.1757	.941	.3072	.892	.3032
12	.972	.1968	.967	.2512	.989	.1649	.936	.3207	.895	.2940
13	.974	.1925	.967	.2491	.991	.1696	.933	.3336	.894	.2960
14	.974	.1867	.971	.2296	.994	.1554	.935	.3227	.915	.2623
15	.970	.1946	.978	.1967	.992	.1517	.944	.3034	.938	.2275
16	.968	.1886	.979	.1958	.991	.1506	.947	.2951	.946	.2137
17	.967	.1907	.979	.1963	.989	.1471	.946	.2962	.942	.2193
18	.965	.1953	.978	.1953	.988	.1503	.946	.2939	.939	.2288
19	.967	.1931	.980	.1900	.990	.1474	.947	.2895	.957	.1879
20	.968	.1950	.980	.1874	.989	.1516	.946	.2926	.948	.2064
21	.968	.1965	.979	.1860	.987	.1565	.946	.2913	.956	.1927
22	.966	.2019	.982	.1824	.992	.1531	.945	.2906	.954	.1927
23	.966	.2019	.982	.1833	.989	.1505	.946	.2899	.955	.1927
24	.961	.2085	.982	.1835	.992	.1560	.945	.2901	.954	.1958
Bst	.975	.1895	.982	.1824	.994	.1471	.947	.2827	.957	.1879

Information Gain

The results of the consecutive modelling using the Information Gain algorithm is presented in Table 4.17. From the table, the MLP model achieved the highest ROC value of 0.997 at 15 features and the lowest RMSE value of 0.1382 at 16 features. The logistic regression classifier model achieved the next best ROC value of 0.982 at 21 features and the J48 classifier achieves the next lowest RMSE value of 0.1814 at 16 features. The naïve Bayes classifier has the least performance with ROC value of 0.948 and RMSE value of 0.2853 at 18 features. The features with the best values for this algorithm lies within 15-16 features. This algorithm achieved the best ROC area value and lowest RMSE value achieved with minimum features of 16 compared to the complete 24 features. This algorithm achieves the purpose of feature selection and it is considered as suitable.

Table 4.17: Performance of the five classifiers on Information Gain ranked attributes

#F	J48		LR		MLP		NB		SMO	
	ROC	RE	ROC	RE	ROC	RE	ROC	RE	ROC	RE
4	.898	.3087	.911	.3120	.931	.2897	.907	.3313	.832	.3617
5	.933	.2415	.933	.2676	.956	.2300	.925	.3126	.896	.2919
6	.932	.2309	.932	.2680	.957	.2255	.919	.3217	.896	.2919
7	.971	.2134	.957	.2686	.982	.1980	.936	.3062	.896	.2877
8	.955	.2334	.959	.2671	.984	.1994	.932	.3013	.894	.2888
9	.952	.2390	.959	.2675	.987	.1968	.927	.3201	.899	.2845
10	.961	.2189	.964	.2590	.989	.1842	.931	.3126	.899	.2845
11	.969	.2060	.967	.2508	.990	.1919	.937	.3029	.895	.2960
12	.974	.1957	.967	.2493	.991	.1660	.933	.3196	.895	.2950
13	.974	.1917	.969	.2326	.993	.1597	.935	.3124	.917	.2587
14	.974	.1867	.971	.2296	.994	.1554	.935	.3227	.915	.2623
15	.974	.1881	.972	.2250	.997	.1390	.935	.3201	.926	.2479
16	.975	.1814	.976	.2140	.997	.1382	.938	.3147	.936	.2314
17	.975	.1834	.976	.2135	.994	.1428	.940	.3081	.937	.2314
18	.967	.1969	.980	.1895	.989	.1464	.948	.2853	.957	.1879
19	.965	.2057	.980	.1874	.988	.1575	.947	.2891	.948	.2064
20	.968	.1950	.980	.1874	.989	.1516	.946	.2926	.948	.2064
21	.966	.2003	.982	.1833	.991	.1468	.946	.2924	.949	.2079
22	.966	.2019	.982	.1824	.992	.1531	.945	.2906	.954	.1927
23	.966	.2019	.982	.1833	.989	.1505	.946	.2899	.955	.1927
24	.961	.2085	.982	.1835	.992	.1560	.945	.2901	.954	.1958
Bst	.975	.1814	.982	.1824	.997	.1382	.948	.2853	.957	.1879

ReliefF

The results of the consecutive modelling using the ReliefF algorithm is presented in Table 4.18. From the table, the MLP model achieved the highest ROC value of 0.995 and the lowest RMSE value of 0.1416 at 20 features. The logistic regression classifier model achieved the next best ROC value of 0.983 at 18 features and the lowest RMSE value of 0.1814 at 20 features. The naïve Bayes classifier has the least performance with ROC value of 0.947 at 19 features and the lowest RMSE value of 0.2883 at 21 features. The range of optimal features for this algorithm is between 18-20 features and this is also considered as unsuitable in terms of achieving the purpose of feature selection. This study considers the ReliefF algorithm as unsuitable for achieving the purpose of feature selection.

Table 4.18: Performance of the five classifiers on ReliefF ranked attributes

#F	J48		LR		MLP		NB		SMO	
	ROC	RE	ROC	RE	ROC	RE	ROC	RE	ROC	RE
4	.906	.2974	.880	.3443	.938	.2837	.875	.3693	.804	.4136
5	.907	.2954	.886	.3412	.944	.2810	.880	.3734	.812	.4016
6	.932	.2684	.914	.3114	.950	.2570	.895	.3591	.863	.3418
7	.942	.2427	.926	.3017	.959	.2205	.911	.3525	.863	.3410
8	.952	.2294	.933	.2856	.972	.1998	.915	.3345	.878	.3101
9	.953	.2295	.933	.2860	.982	.1913	.908	.3533	.875	.3140
10	.950	.2117	.956	.2396	.985	.1807	.928	.3126	.923	.2552
11	.943	.2155	.956	.2403	.983	.1617	.928	.3135	.921	.2516
12	.948	.2177	.961	.2286	.989	.1501	.931	.3079	.930	.2405
13	.946	.2198	.963	.2260	.991	.1523	.927	.3194	.930	.2392
14	.952	.2137	.968	.2119	.985	.1609	.929	.3181	.935	.2366
15	.953	.2149	.969	.2102	.982	.1611	.930	.3199	.940	.2314
16	.961	.2081	.979	.1984	.990	.1599	.942	.3087	.944	.2207
17	.963	.2047	.981	.1887	.992	.1418	.943	.3065	.950	.2064
18	.962	.2043	.983	.1829	.994	.1506	.946	.2951	.951	.2034
19	.962	.2042	.982	.1822	.991	.1485	.947	.2915	.953	.1989
20	.966	.2054	.983	.1814	.995	.1416	.945	.2921	.955	.1927
21	.966	.2052	.982	.1824	.994	.1453	.946	.2883	.954	.1927
22	.966	.2019	.982	.1824	.992	.1531	.945	.2906	.954	.1927
23	.961	.2085	.982	.1825	.993	.1482	.945	.2908	.954	.1958
24	.961	.2085	.982	.1835	.992	.1560	.945	.2901	.954	.1958
Bst	.966	.2019	.983	.1814	.995	.1416	.947	.2883	.955	.1927

4.3.1.1 Summary of Results

This section discusses the performance summary of the feature selection algorithms used to obtain the best features for this study. The algorithms used for this study are Correlation, Gain Ratio, Information Gain and ReliefF. The Table 4.19 showing the results summary presents the highest ROC value, the lowest RMSE value achieved by each algorithm and the range of the best features. These values were obtained from the successive modelling of each algorithm's ranked features starting from the top four features until all the 24 features were modelled. From the table, the Information Gain algorithm performed best with minimum features achieving the highest ROC value of 0.997 and the lowest RMSE value of 0.1382.

Table 4.19: Performance summary of feature selection algorithms used for selecting the best features

Algorithm	Highest ROC value	Lowest RMSE value	Range of best features
Correlation	0.993	0.1410	15-21
Gain Ratio	0.994	0.1471	14-17
Information Gain	0.997	0.1382	15-16
ReliefF	0.995	0.1416	18-20

The summary of the performance of the feature selection algorithms show that minimum features can perform better than complete features because results from all the algorithms show a higher level of performance with less features than with the complete features. The model with the best performance employing the Information Gain algorithm is the multilayer perceptron classifier and its performance with minimum features between the ranges of 15-16 features has ROC value of 0.997 and RMSE value of 0.1382 compared with the complete features where the ROC value is 0.992 and RMSE value is 0.1560. This shows that the minimum features perform better than the complete features. Therefore, this study discards the least 8 ranked features and implements the top 16 ranked features with the Information Gain algorithm in the design of the prediction application for the Niger Delta University. The 16 selected features for the prediction application are: *Sponsor qualification, Weekly study time, Average SSCE score, Sponsor type, Jamb score, Sponsor support, Secondary school type, Work and study, Sponsor income, Secondary school area, University accommodation, Postgraduate degree, Years before admission, Sports activeness, Post-UTME score and Course interest.*

4.3.2 Performance of Multilayer Perceptron Classifier using the Best Selected Features

The results from the summary of the feature selection algorithms in Section 4.3.1.1 shows that the Information Gain algorithm using the multilayer perceptron classifier achieved the best performance with minimum attributes of the top 16 ranked attributes. This section builds a new model for the study with the 16 selected attributes and the multilayer perceptron classifier, which achieves the best values from the consecutive modelling process. The modelling process begins with extracting the selected attributes in WEKA and then making use of the multilayer perceptron classifier to build the model. The Table 4.20 presents the results summary of the training dataset for the model built with the MLP classifier and the best features while the diagram in Fig 4.17 shows the model built in WEKA environment.

Table 4.20: Summary of multilayer perceptron performance results using the best features dataset with the training dataset

Classification Items	Values	Metrics	Values
Correctly classified students	1606	Recall	0.985
Incorrectly classified students	37	Specificity	0.965
Correctly classified HL students	579	ROC	0.997
Incorrectly classified HL students	21	F-Measure	0.982
Correctly classified LL students	1027	Kappa	0.9513
Incorrectly classified LL students	16	RMSE	0.1382

From the table, all the metric values with the best features show good performance and when compared with the values obtained with the complete features; the best features perform even better. In comparison between the metrics values for the best features and the complete features; Recall for best features is 98.5% while Recall for complete features is 98.3%; Specificity for best features is 96.5% while Specificity for complete features is 95.2%; ROC for best features is 99.7% while ROC for complete features is 99.2%. The F-Measure for best features is 98.2% while F-Measure for complete features is 97.8%; Kappa for best features is 0.9513 while Kappa for complete features is 0.9381; RMSE for best features is 0.1382 while RMSE for complete features is 0.1560. The results show improvement for every metric value using the 16 best features. From the results obtained with the minimum features, this study concludes that the multilayer perceptron classifier and the 16 optimal features ranked with the Information Gain algorithm is the best classifier for the dataset used for this study. Therefore, it is beneficial to design and implement the prediction application for the Niger Delta University with the multilayer perceptron classifier and the 16 selected features.


```

Time taken to build model: 6.92 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      1606           97.748 %
Incorrectly Classified Instances     37            2.252 %
Kappa statistic                    0.9513
Mean absolute error                 0.0269
Root mean squared error             0.1382
Relative absolute error             5.7956 %
Root relative squared error         28.7039 %
Total Number of Instances          1643

=== Detailed Accuracy By Class ===

                TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                0.965   0.015   0.973     0.965   0.969     0.951   0.997    0.995    HL
                0.985   0.035   0.980     0.985   0.982     0.951   0.997    0.998    LL
Weighted Avg.   0.977   0.028   0.977     0.977   0.977     0.951   0.997    0.997

=== Confusion Matrix ===

  a    b  <-- classified as
579  21 |    a = HL
 16 1027 |    b = LL

```

Fig 4.17: The multilayer perceptron model built with the best features using the training dataset

After obtaining the results from the training dataset from the MLP classifier using the best 16 features, the study tests the model with the test dataset to ensure that the model is also free from bias and can generalise well. The 16 selected attributes were extracted from the test dataset and used to test the model designed using the best features with the MLP classifier. The Table 4.21 presents the performance results of the testing process and the diagram in Fig 4.18 shows the model in WEKA environment.

Table 4.21: Summary of multilayer perceptron performance results using the best features dataset with the test dataset

Classification Items	Values	Metrics	Values
Correctly classified students	691	Recall	0.993
Incorrectly classified students	14	Specificity	0.957
Correctly classified HL students	245	ROC	0.996
Incorrectly classified HL students	11	F-Measure	0.985
Correctly classified LL students	446	Kappa	0.9568
Incorrectly classified LL students	03	RMSE	0.1323

```

=== Re-evaluation on test set ===

User supplied test set
Relation:      NewDataset2-weka.filters.unsupervised.instance.Resample-S1-Z70.0-no-replacement-V-weka.filters.
Instances:     unknown (yet). Reading incrementally
Attributes:    17

=== Summary ===

Correctly Classified Instances      691          98.0142 %
Incorrectly Classified Instances    14           1.9858 %
Kappa statistic                    0.9568
Mean absolute error                 0.0236
Root mean squared error             0.1323
Total Number of Instances          705

=== Detailed Accuracy By Class ===

          TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
          0.957   0.007   0.988     0.957   0.972     0.957   0.996     0.994     HL
          0.993   0.043   0.976     0.993   0.985     0.957   0.996     0.997     LL
Weighted Avg.   0.980   0.030   0.980     0.980   0.980     0.957   0.996     0.996

=== Confusion Matrix ===

  a  b  <-- classified as
245 11 |  a = HL
  3 446 |  b = LL

```

Fig 4.18: The multilayer perceptron model obtained with the best features using the test dataset

From the table, the metrics for evaluation obtained good values with the test dataset and in comparison with the training dataset, it shows that the model performs well and can generalise competently with real data. The metrics values obtained with the training dataset compared to the test dataset are Recall for training dataset is 98.5% while Recall for test dataset is 99.3%; Specificity for training dataset is 96.5% while Specificity for test dataset is 95.7%; ROC for training dataset is 99.7% while ROC for test dataset is 99.6%. The F-Measure value for training dataset is 98.2% while F-Measure for test dataset is 97.8%; Kappa for training dataset is 0.9513 while Kappa for test dataset is 0.9568; RMSE for training dataset is 0.1323 while RMSE for test dataset is 0.1560. The good performance obtained using the model built for testing shows that the model performs very well with the test dataset and can generalise well.

4.4 Chapter Summary

This chapter presented the results and discussion from the EDM process using the WEKA modelling tool. The chapter modelled the dataset collected for the research by applying five machine learning algorithms namely, J48, logistic regression, multilayer perceptron, naïve Bayes and sequential minimal optimization. Before modelling, the dataset was split into two parts, one part for training the model and the second part for testing the model built with the trained dataset.

The presentation, interpretation and performance evaluation of the training and test dataset were presented in this chapter. From the modelling process, the multilayer perceptron classifier was deemed the best model for classifying students' performance for the dataset used in this study.

The chapter further strove to select the best features within the dataset by using four feature selection algorithms called Correlation, Gain Ratio, Information Gain and ReliefF. The results obtained from the algorithms ranking features in order of importance from most important led to the consecutive modelling of features starting from the top four until all the complete features were modelled. The consecutive modelling process enabled this research to clarify that with the 16 most important attributes ranked by the Information Gain algorithm and the multilayer perceptron classifier achieves its best performance. Therefore, the study conclusively used the selected 16 attributes and the multilayer perceptron classifier in the design and implementation of the prediction application for the Niger Delta University. The next chapter examines and extrapolates the development process of the prediction application for Niger Delta University using the 16 attributes and the multilayer perceptron model.

CHAPTER FIVE: DESIGN, IMPLEMENTATION AND EVALUATION OF PREDICTIVE SYSTEM

5.1 Introduction

This study proposed to design a predictive system that models the features of low performing undergraduate students in NDU. Through this model, new students with similar characteristics can be identified early and the university can set up the essential interventions to cater for these students' needs. To achieve this aim, the research looked at five machine-learning classifiers and used the data collected from NDU to build models using the WEKA modelling tool. The previous chapter presented the results from the modelling process.

This chapter develops the final stage of the research methodology (CRISP-DM) of this study, which is the deployment stage. For this deployment stage, the research developed a novel software for identifying low academic performance using the best algorithm identified in the modelling phase. To achieve the design of the predictive application for deployment, the research follows the design approach of gathering relevant requirements for the software, designing and evaluating the system.

5.2 The Study Perspective

The use of technology provides easy ways to carry out tasks and produce accurate and timely results, thus different sectors seek technological methods to enhance their productivity. In education, the use of data mining technology has shown developments by the design of models, behavioural patterns and tools for educational data such as decisional tool (Selmoune & Alimazighi, 2008), LiMS (Macfadyen & Sorenson, 2010), EDM Workbench (Rodrigo et al, 2012), gaming the system (Baker et al, 2004), Off-task behaviour (Baker, 2007), Without Thinking Fastidiously (WTF) behaviour (Wixon et al, 2012). All of these models and systems have been developed and applied in few educational institutions, mostly in developed countries (Guri-Rosenblit, 2006) and this constrains the growth and development of the educational data mining community in developing countries.

To promote the development of EDM in developing countries, this research adds to the body of knowledge in EDM by designing a novel computer-based system that helps identify and classify low performing students into failure risk levels. The system helps to determine students that require

intervention early enough and with the results from the system students can receive the necessary assistance they need to progress in their academics.

5.2.1 Components of the Predictive System

This section offers a brief discussion of the components of the predictive system, which are the client and server. The client component of the predictive system is the web application designed with php programming language, which connects to the server through the internet. The client component presents a menu for entry of students' data, which contains the best 16 features identified in section 4.3.1.1 by the feature selection technique during the data mining process and the server component of the predictive system contains the multilayer perceptron classifier model implemented in php using machine-learning libraries.

5.2.2 The Design Process

The design process followed in the design of the predictive system for NDU commenced with the selection of the best classifier model for this study presented in Chapter 4. The algorithm below represents the design process derived from the requirements for the predictive system.

Algorithm 5.1: The design process

- Step 1. Start
- Step 2. Input student's features, which are the best features identified (*Sponsor qualification, Weekly study time, Average SSCE score, Sponsor type, Jamb score, Sponsor support, Secondary school type, Work and study, Sponsor income, Secondary school area, University accommodation, Postgraduate degree, Years before admission, Sports activeness, Post-UTME score and Course interest*)
- Step 3. Use the best classifier identified, which is the multilayer perceptron model
- Step 4. Predict student's failure risk as HL or LL
- Step 5. Output summary of invention required for student, student with HL (low risk) failure requires low intervention and student with LL (high risk) failure requires high intervention
- Step 6. Stop

5.2.3 The System Requirements

In the organization of the Niger Delta University, each faculty within the university has faculty officers in charge of collecting and archiving students' information. These faculty officers are the users of the prediction software designed and developed in this study. The role they play is to use the prediction application, get the predictions and forward the results to relevant authorities within the faculty. This initial processing ensures that intervention measures are set up early enough for the students correctly predicted by the software. From this premise, the research deduced the following list of requirements for the prediction software

1. The application interface must be simple and easy to use
2. The system must be secured and allow only authorised persons to make use of it
3. The system must prevent error in prediction by ensuring all fields are selected
4. The system must provide results timely and accurately
5. The system must predict students risk level to enable early intervention

5.2.4 Sample Model

Prior to the development of the prediction application, the study designed a sample or pilot model of the software. This sample model designed using the storyboard plugin in Microsoft PowerPoint application shows a visual picture of the proposed software and UML diagrams to show the functions and activities of the proposed software. Based on this visual sample, it was easy to see how the proposed predictive system would behave and thus helped in creating the interface design of the predictive software easy. The diagrams presented in Fig 5.1 and Fig 5.2 amplify the UML diagrams while Fig 5.3 – Fig 5.6 demonstrates the design of the sample model.

5.2.4.1 Sample model design

This section offers sample model designs, which include the visual picture of the software using storyboard and some system designs such as the use case and context diagrams. A use case diagram illustrates users' interaction with the system by showing the connection between users (or actors) of the system and their roles (or actions) with the system (Sengupta & Bhattacharya, 2006). Fig 5.1 shows the case diagram adapted and contextualised for this research. A context diagram shows the relationship between the system and other external entities such as external data storage, users and other external systems (Sommerville, 2011). The context diagram shows the boundaries of the software and other systems that communicate with it.

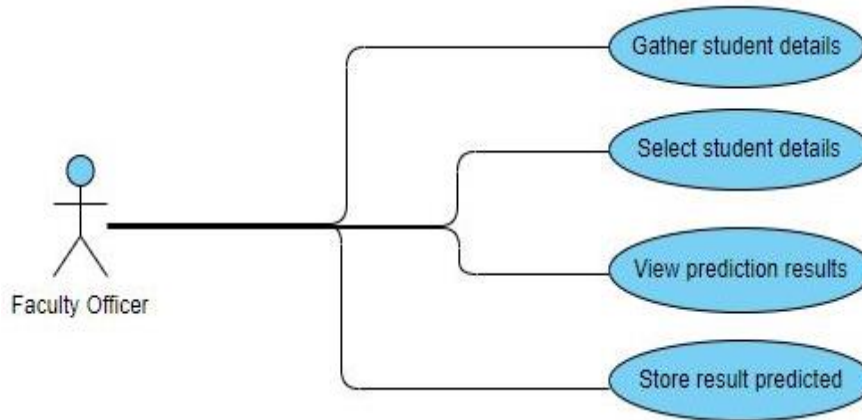


Fig 5.1: Use case diagram showing the Faculty Officer’s roles in using the predictive system

The use case diagram presented in Fig 5.1 shows the main actor or user of the system as the faculty officer. Only faculty officers primarily use the system as they are responsible for gathering and storing of student data, thus they are in the best position to use the system and communicate results with relevant stakeholders. The activities performed by the actors as shown in the diagram include gathering of students’ details, selecting student details in the software, viewing the predicted results and storing the predicted results for future use.

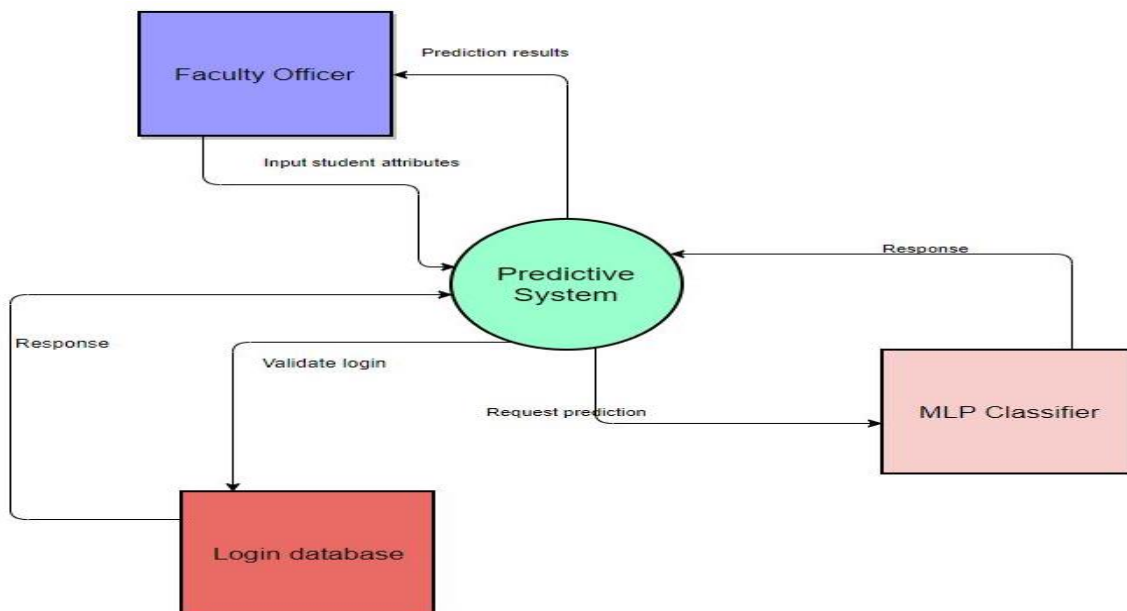


Fig 5.2: Context diagram showing the data process and flow within the system

The diagram in Fig 5.2 presents the context diagram of the software. This diagram shows the boundary of the system and other systems that interact with it. At the centre of the diagram is the predictive system and the faculty officer, login database and MLP classifier are systems that

interact with the predictive system. From the diagram, the Faculty Officer communicates with the predictive system by inputting student details and receiving the result predicted from the predictive system. The login database holds the login details from faculty officers and interacts with the predictive system by receiving requests to validate login details and it gives response of successful or unsuccessful validation. The MLP classifier receives request to predict student performance based on students' details given and responds with the corresponding prediction for the students.

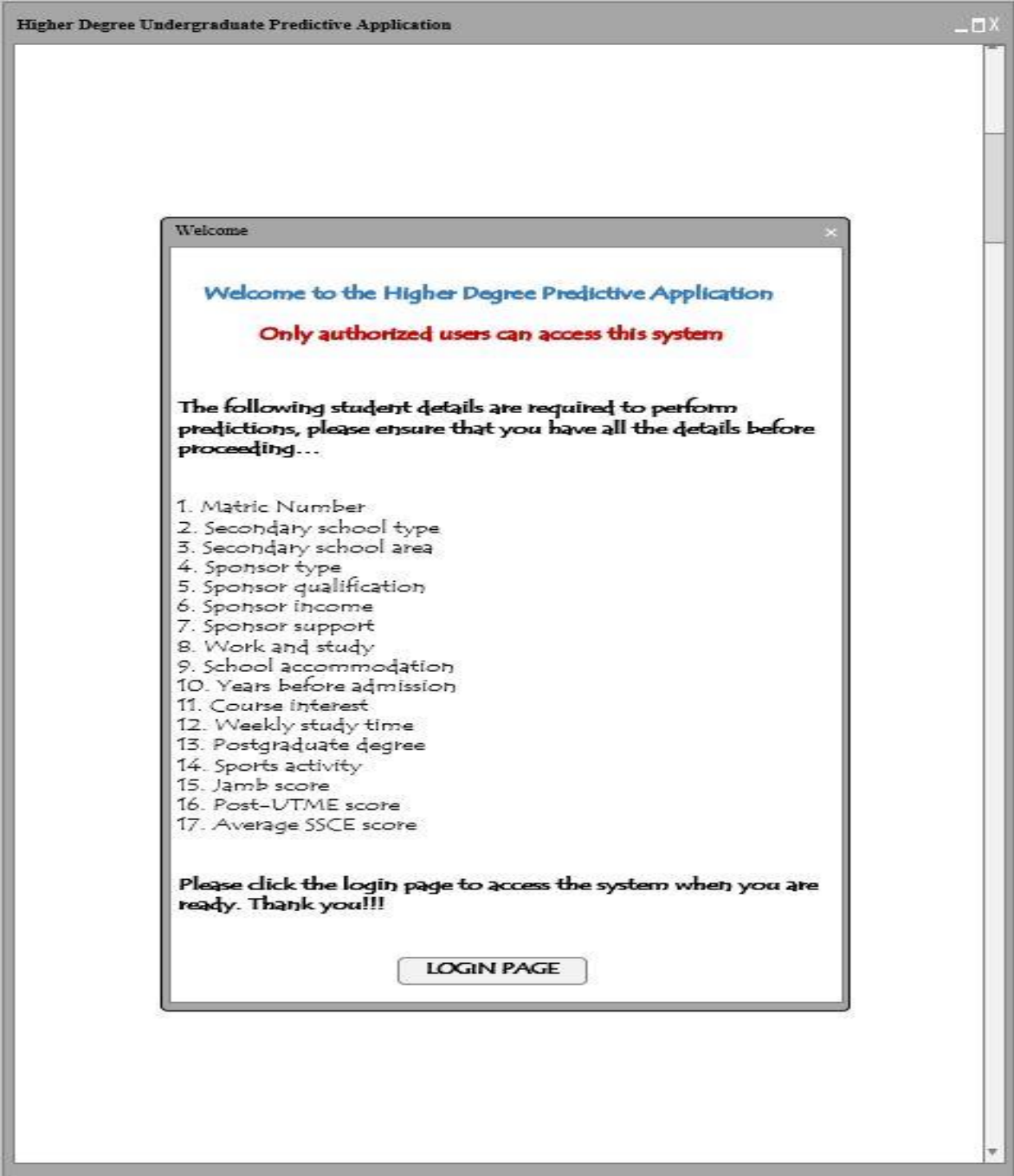


Fig 5.3: Welcome screen

The diagram in Fig 5.3 shows the storyboard design of the welcome screen. This indicates the first page of the proposed web application, which offers instruction to users of the system on relevant student details to gather for the prediction purpose. This screen also informs the user that only authorized persons can access the software and has a login button to help the user gain access to the system. This page is relevant as it helps the system achieve a user-friendly feature and assist users know the relevant data to gather before using the system.

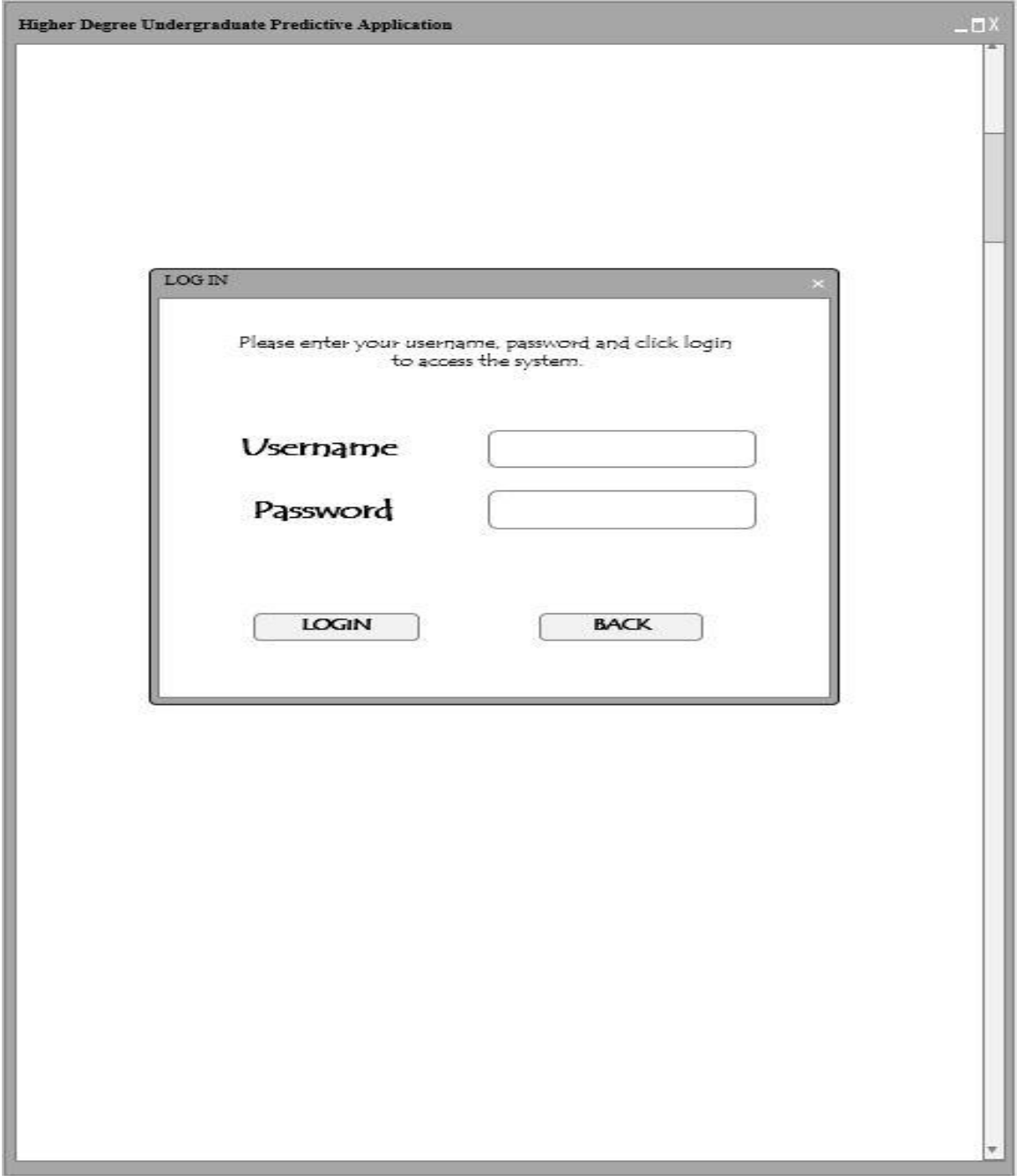


Fig 5.4: Login Screen

The diagram in Fig 5.4 shows the storyboard design of the login screen. This page of the proposed web application allows users input their login details. These details are stored in a database, thus the details are first verified by checking the database to ensure that the user’s details, which are username and password, are stored there. Upon successful verification, the system grants the user access to the software.

Higher Degree Predictive Application Input

Please select all fields and click view result to view the result predicted for the student.
All fields are required for the processing of the results.

1. Matric Number: UG/000/0000
2. Secondary school type: Private
3. Secondary school area: Rural
4. Sponsor type: Guardian
5. Sponsor qualification: Degree
6. Sponsor income: High
7. Sponsor support: High
8. Work and study: No
9. University accommodation: Campus
10. Years before admission: Before five years
11. Course interest: High
12. Weekly study time: Average
13. Postgraduate degree interest: No
14. Sports activeness: High
15. Jamb score: High
16. Post-UTME score: High
17. Average SSCE score: Average

View Result Back

Fig 5.5: Sample design of predictive application for student information form

The diagram in Fig 5.5 depicts the storyboard for the webpage of the proposed software. This diagram shows the 16 features selected as the best features determined by the feature selection

technique in section 4.3.1.1 and an additional feature called Matric Number for identifying the student. This page shows that the user of the system can select the corresponding options for a student from the dropdown menu of each feature. It is important to note that all attributes must be selected for the system to offer predictions; therefore, the software must provide some measure to ensure that this is achieved. The page also shows two buttons for viewing the predicted results and for resetting the application.

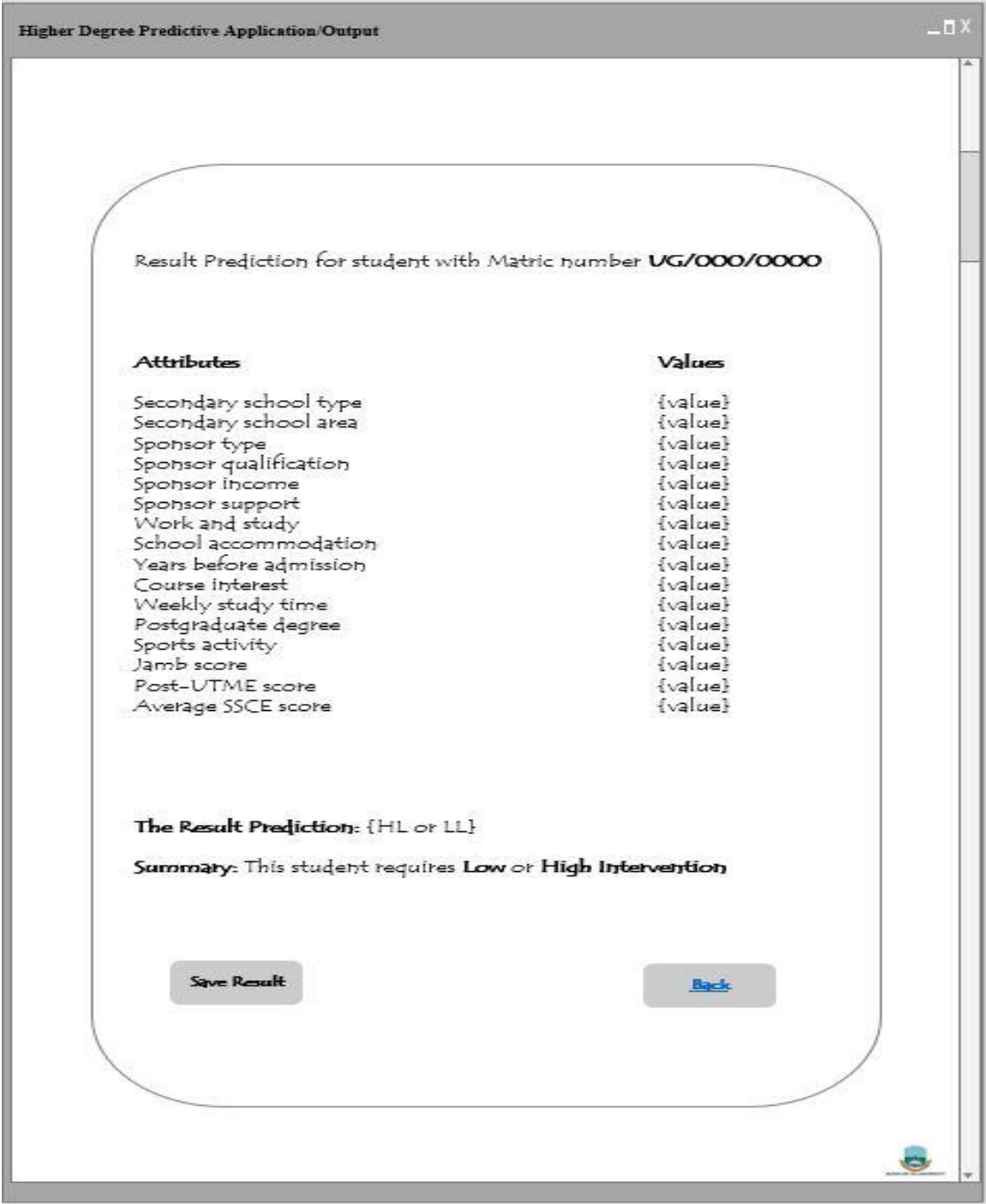


Fig 5.6: Sample design of predictive application for result prediction

The diagram in Fig 5.6 shows the storyboard of the result predicted for a student using the Matric Number to identify the student. This diagram illustrates each attribute in the system and the value of the attribute selected for the student, the “Result Prediction” label shows the class predicted by the software using the multilayer perceptron algorithm, which can be HL for low risk students and LL for high-risk students. Finally, the “Summary” label describes the level of intervention required for the student, which is either high intervention or low intervention; students with high-risk level requires high intervention and students with low risk level requires low intervention.

With this sample model design created, the research easily designed the interface of the software; the next section discusses the predictive system design process and methods used to achieve the design.

5.3 Prototype of the Predictive System

In the development of software, it is important to follow a software development methodology that clarifies in simple terms the entire development process of the software. For this research, the development of the predictive system follows rapid prototyping methodology described in section 3.3.5.1 for the design of the software.

The sample model in Section 5.3.2 assisted in the design of the system interface for the predictive software using the php programming language. The diagrams Fig 5.8 to Fig 5.11 summarise and amplify the snapshots of the designed application.

5.3.1 Description of the predictive software design

This section provides a thick description of the steps followed in the design of the predictive system. In order to achieve this, this research first outlines the use of the model built in WEKA to achieve the prediction. Many languages incorporate the WEKA modelling tool and one of such language is Python. The steps below describe the use of the WEKA model, python and php programming languages in the software:

1. To use weka functionality in python the study makes use of wekapy library

```
from wekapy import *
```

Wekapy allow us to load the model created by WEKA software.

It is vital to mention the “classifier_type” which in this case in the MLP when loading the model.

```
model = Model(classifier_type = "functions.MultilayerPerceptron")  
model.load_model("./MLP model.model")
```

2. After loading the model, next is to create an instance of the model by using the instance function, which can assist with adding input parameter to the model.

```
test_instance1 = Instance()
```

3. The model consists of input and output parameters. Input parameters are the inputs required by the model to predict the output. In the case of this study the input parameters/features are the best selected 16 features given below

```
Feature(name="SecType", value=schoolType, possible_values="{Pri, Pub}"),  
Feature(name="SecArea", value=schArea, possible_values="{Urb, Rur}"),  
Feature(name="SponType", value=sponsType, possible_values="{Guad, Par, Self}"),  
Feature(name="SponQual", value=sponsQual,  
possible_values="{Deg, NoDeg, NoEdu}"),  
Feature(name="SponInc", value=sponsInc, possible_values="{Med, High, Low}"),  
Feature(name="SponSup", value=sponsSup, possible_values="{High, Med, Low}"),  
Feature(name="WorkStudy", value=workStud, possible_values="{No, Yes}"),  
Feature(name="UniAcc", value=uniAcc, possible_values="{Cmps, OffCmps}"),  
Feature(name="BeAdmYrs", value=yearsBefore, possible_values="{None, 5A, B5}"),  
Feature(name="CouInt", value=courseInt, possible_values="{High, Ave, Low}"),  
Feature(name="WkStud", value=studTime, possible_values="{High, Ave, Low}"),  
Feature(name="PgDeg", value=postDeg, possible_values="{Yes, Ns, No}"),  
Feature(name="SptAc", value=sportAct, possible_values="{High, Low}"),  
Feature(name="JambSc", value=jambScore, possible_values="{High, Low, Ave}"),  
Feature(name="PumeSc", value=pumeScore, possible_values="{Ave, Low, High}"),  
Feature(name="AveSc", value=ssceScore, possible_values="{Ave, High, Low}"),
```

The output parameter is the prediction and its value is unknown (signified by question mark) for every new student

```
Feature(name="CGPA", value="?", possible_values="{HL,LL}")
```

4. After setting the input and output features, the study adds the created instance to the model; this helps to predict the output from the model.

```
model.add_test_instance(test_instance1)  
model.test()
```

5. For hosting this code as a web service the study makes use of Flask which a python framework for microservice. The code for using Flask is given below

```
from flask import Flask,request, jsonify  
import json  
app = Flask(__name__)  
@app.route("/", methods=['POST','PUT'])  
def predict():
```

6. Using flask, the study fetches the input parameters from the php website to model the service with the code below:

```
field_mlp = request.json  
field_mlp_dict = dict(field_mlp)
```

7. The code below fetches the data from the PHP website

```
schoolType = field_mlp_dict['schoolType']  
schArea = field_mlp_dict['schArea']  
sponsType = field_mlp_dict['sponsType']  
sponsQual = field_mlp_dict['sponsQual']  
sponsInc = field_mlp_dict['sponsInc']  
sponsSup = field_mlp_dict['sponsSup']
```

```

workStud = field_mlp_dict['workStud']
uniAcc = field_mlp_dict['uniAcc']
yearsBefore = field_mlp_dict['yearsBefore']
courseInt = field_mlp_dict['courseInt']
studTime = field_mlp_dict['studTime']
postDeg = field_mlp_dict['postDeg']
sportAct = field_mlp_dict['sportAct']
jambScore = field_mlp_dict['jambScore']
pumeScore = field_mlp_dict['pumeScore']
ssceScore = field_mlp_dict['ssceScore']

```

8. When the model predicts the output, we return the prediction to php website and that result is displayed to the user

```

predictions = model.predictions
predict = str(predictions[0])
return predict[26:28]

```

9. Finally, the study makes use of the code below to host the python file as a web service

```

app.run(host = '0.0.0.0',port = 5011, debug = False, threaded=True)

```

5.3.2 Prototype model design

The design of the interface for the predictive system makes use of php programming language, MySQL database server for storing user login details. XAMPP is web application that allows easy testing and deployment of software, both php and MySQL as components included in the XAMPP software. This section presents snapshots of the designed predictive application. It focuses on the welcome screen for offering more information to users of the system on features required to use the software. It then moves to the login screen showing the screen where users input their login details. The predictive application screen for users to input the features required for prediction follows and then finally the result screen projects the output of the prediction based on input received.

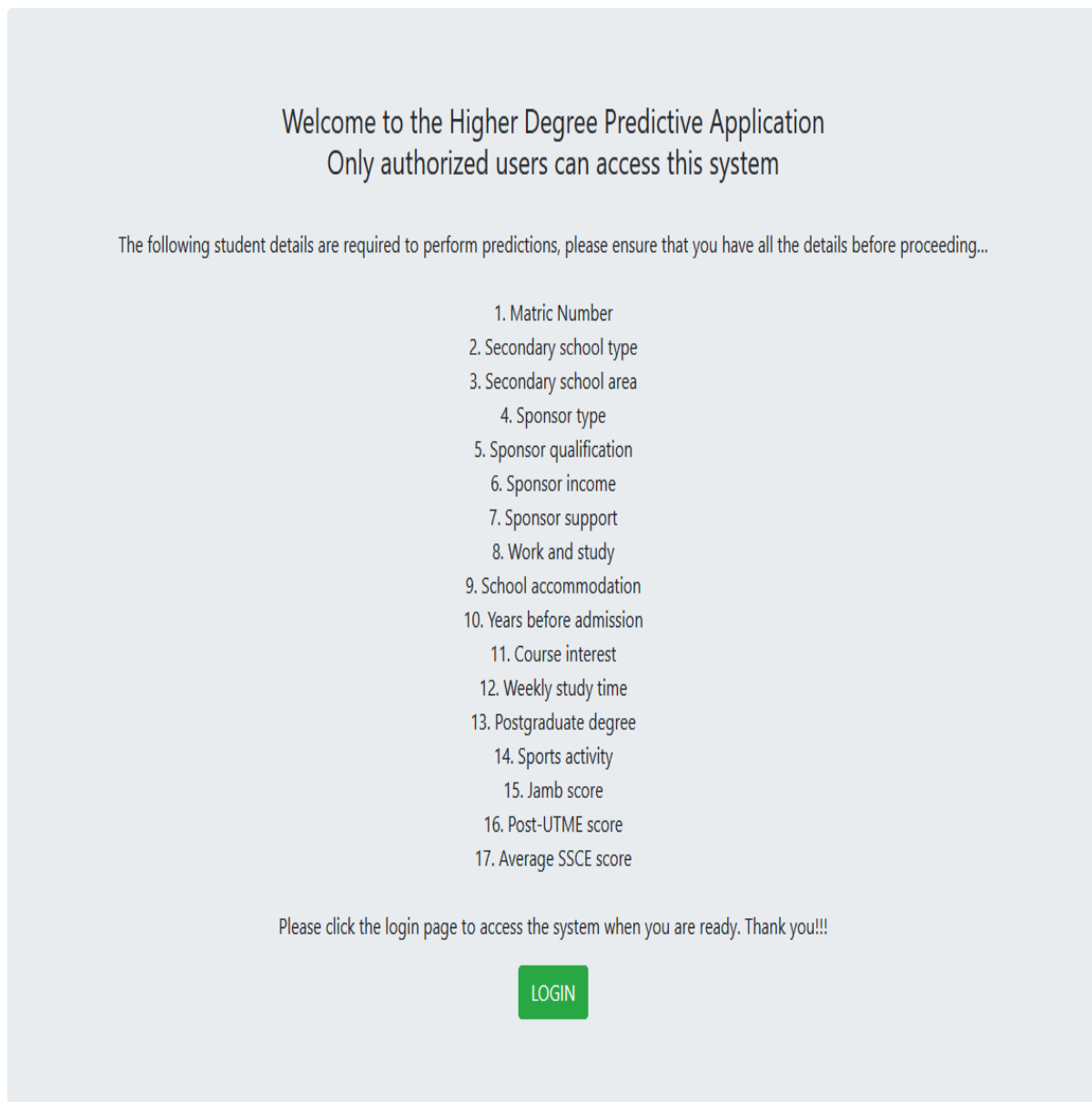


Fig 5.7: Welcome Screen

The snapshot in Fig 5.8 shows the design of the welcome screen. This is the first page of the web application that offers instruction to the system user on the relevant student details. This screen also informs the user that only authorized persons can access the software: this page also has a login button to allow the user gain access to the system. On this screen, the user receives prompt information of features required to use the system and this makes the software user friendly and interactive; user-friendliness and interactivity are part of the requirements of the software. The initial security feature notifies users to login before they can gain access to use the software.

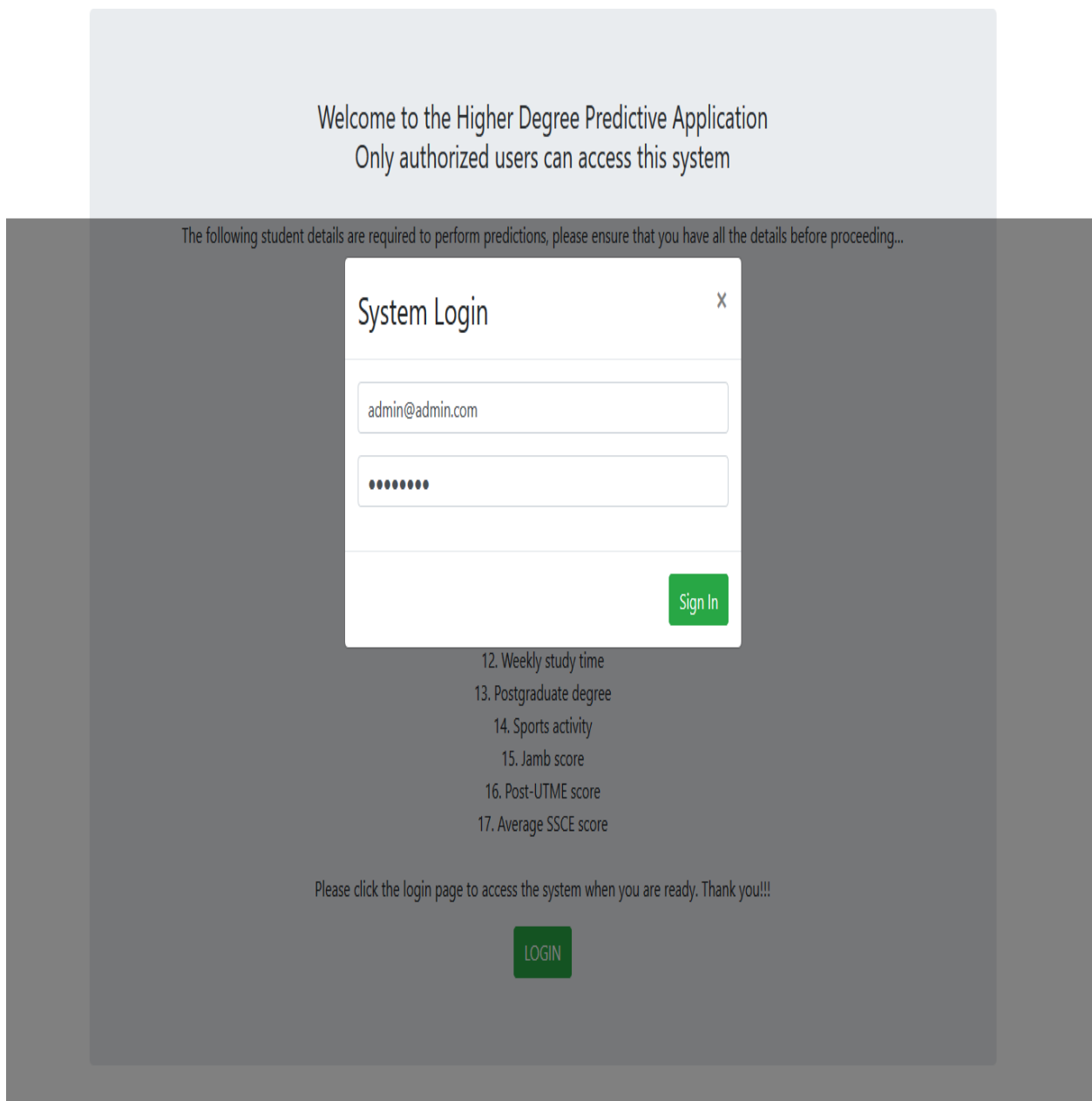


Fig 5.8: System Login Page

The snapshot in Fig 5.9 shows the design of the login screen. This page of the web application allows users to input their login details. These details are verified by the database to ensure that the user's details, which are username and password, are stored appropriately and are activated immediately the system recognises these credentials. Upon successful verification, the system grants the user access to the software. This screen provides security for the software, thereby ensuring that only authorised persons can gain access to the system. The MySQL database server contains the login details of all authorised persons.

Please select all fields and click **view result** to view the result predicted for the student. All fields are required for the processing of the results.

Matric Number
 UG/018/1789

Secondary School Type
 PUB

Secondary School Area
 RUR

Sponsor Type
 SELF

Sponsor Qualificattion
 NODEG

Sponsor Income
 LOW

Sponsor Support
 MED

Work And Study
 YES

University Accommodation
 OFFCMPS

Years Before Admission
 5A

Course Interest
 AVE

Weekly Study Time
 LOW

Post Graduate Degree Interest
 NS

Sport Activeness
 LOW

Jamb Score
 AVE

Post-UTME Score
 LOW

Average SSCE Socre
 LOW

View Result **Reset**

Fig 5.9: Prediction Application showing the user input

The snapshot in Fig 5.10 depicts the input webpage of the software that shows the 16 features selected as the best. These are determined by the feature selection technique and an additional feature called Matric Number for identifying the student. This page shows that the user of the system can select the corresponding options for a student from the dropdown menu of each feature. The dropdown menu helps to rid the system of typing errors as all possibilities are predetermined. The screen also shows two buttons for viewing the predicted results and for resetting the webpage.

Result prediction for **UG/018/1789**

Attributes	Values
Secondary School Type	Public
Secondary School Area	Rural
Sponsor Type	Self Sponsor
Sponsor Qualificattion	Educated without Degree
Sponsor Income	Below N50,000
Sponsor Support	Average support
Work and Study	Yes
University Accommodation	Off Campus
Years before Admission	5 years and above
Course Interest	Average interest
Weekly Study Time	Less than 10hrs
Post Graduate degree Interest	Not Sure
Sport Activeness	A little Active
Jamb Score	180 – 250
Post-UTME Score	Below 180
Average SSCE socre	Less than 4.00

Result prediction: LL

Summary: Student requires **Student requires high Intervention**

Save Result Back

Fig 5.10: Prediction Application showing the prediction results

The snapshot in Fig 5.11 shows the result predicted for a student using the Matric Number to identify the student. This diagram illustrates each attribute in the system and the value of the attribute selected for that student, the “Result Prediction” label shows the class predicted by the software, which can be HL for low risk students and LL for high-risk students. This result prediction is possible by the use of the multilayer perceptron algorithm model built in WEKA combined with the python and php programming languages as described in section 5.4.1. The “Summary” label describes the level of intervention required for each student based on the

prediction output; thus, students with high-risk level (LL students) require high intervention and students with low risk level (HL students) require low intervention.

5.4 Evaluation of the Predictive System

This section discusses the evaluation of the system based on the requirements gathered, which helps to confirm that the designed software meets the system requirements.

5.4.1 Software Evaluation

This section presents an evaluation of the designed predictive software. The complete dataset for this research involved 2348 student records collected from the Niger Delta University. From the dataset, 30% set aside for testing using the resample filter in WEKA (details in Chapter Four) contains 705 student records. The snapshot in 5.12 depicts a section of the test dataset in an Excel file showing student records, actual result for the student and the predicted result acquired from the software.

MatNum	SecType	SecArea	SponType	SponQual	SponInc	SponSup	WorkStud	UniAcc	BeAdmYrs	Count	WkStud	PgDeg	SptAc	JambSc	PumeSc	AveSc	Actual	Predicted	LowLow	HighLow
UG/010/0001	Pub	Rur	Self	NoDeg	Med	Med	Yes	OffCmps	5A	Ave	Ave	Ns	Low	Low	Ave	Ave	LowLow	LL	A	
UG/010/0002	Pub	Rur	Self	NoDeg	Med	Low	Yes	OffCmps	5A	Ave	Ave	Ns	Low	Low	Low	Low	LowLow	LL	A	
UG/010/0003	Pub	Rur	Par	NoDeg	Med	High	No	OffCmps	B5	Ave	Ave	Ns	Low	Ave	High	Ave	LowLow	LL	A	
UG/010/0004	Pri	Urb	Guad	Deg	High	High	No	OffCmps	B5	High	Ave	Yes	Low	High	Ave	High	HighLow	HL		A
UG/010/0005	Pri	Rur	Self	NoDeg	Med	Low	Yes	OffCmps	5A	Ave	Low	Ns	Low	Low	Ave	Ave	LowLow	LL	A	
UG/010/0006	Pri	Rur	Guad	Deg	High	Med	Yes	Cmps	5A	Ave	High	Ns	High	Low	Low	Low	HighLow	HL		A
UG/010/0007	Pri	Rur	Self	NoDeg	Med	Low	Yes	OffCmps	B5	Ave	Low	Ns	High	Low	Ave	Low	LowLow	HL	D	
UG/010/0008	Pub	Rur	Self	NoDeg	Med	Low	Yes	Cmps	5A	Ave	Ave	Ns	High	Ave	Ave	Ave	LowLow	LL	A	
UG/010/0009	Pub	Rur	Self	NoDeg	Low	Low	Yes	OffCmps	5A	Ave	Low	Ns	Low	Low	Low	Low	LowLow	LL	A	
UG/010/0010	Pub	Urb	Guad	Deg	High	High	No	OffCmps	B5	Ave	Ave	Ns	High	Ave	High	High	HighLow	HL		A
UG/010/0011	Pub	Urb	Guad	Deg	High	High	No	OffCmps	B5	Ave	Ave	Ns	High	Ave	High	High	HighLow	HL		A
UG/010/0012	Pri	Rur	Self	NoDeg	Med	Med	Yes	OffCmps	5A	Low	Ave	Ns	Low	High	Ave	Ave	LowLow	LL	A	
UG/010/0013	Pub	Urb	Guad	Deg	High	High	No	Cmps	B5	High	High	Yes	High	Ave	High	High	HighLow	HL		A
UG/010/0014	Pri	Urb	Guad	Deg	High	High	No	Cmps	B5	High	High	Yes	High	Ave	High	High	HighLow	HL		A
UG/010/0015	Pri	Rur	Guad	Deg	High	Med	Yes	Cmps	5A	Ave	High	Ns	High	Low	Low	Ave	HighLow	HL		A
UG/010/0016	Pri	Urb	Guad	Deg	High	High	No	OffCmps	B5	High	Low	Yes	Low	Ave	Low	Ave	HighLow	HL		A
UG/010/0017	Pub	Rur	Self	NoDeg	Med	Med	Yes	OffCmps	5A	Ave	Ave	Ns	Low	High	Ave	Ave	LowLow	LL	A	
UG/010/0018	Pub	Rur	Self	NoDeg	Med	Med	Yes	Cmps	5A	Ave	High	Ns	High	Low	Low	Ave	LowLow	LL	A	
UG/010/0019	Pub	Rur	Guad	NoDeg	Low	Low	No	OffCmps	B5	Ave	Low	Ns	Low	Low	Low	Low	LowLow	LL	A	
UG/010/0020	Pri	Urb	Guad	Deg	High	Med	No	Cmps	B5	High	High	Yes	High	Ave	High	Ave	HighLow	HL		A
UG/010/0021	Pub	Urb	Self	NoDeg	Med	Low	Yes	OffCmps	B5	High	High	Ns	Low	Ave	Ave	Ave	LowLow	LL	A	
UG/010/0022	Pub	Rur	Par	NoDeg	Med	High	No	OffCmps	B5	Ave	Low	Ns	Low	Ave	Low	Ave	LowLow	LL	A	
UG/010/0023	Pri	Urb	Guad	Deg	High	High	No	Cmps	None	High	High	Yes	High	High	Ave	Ave	HighLow	HL		A
UG/010/0024	Pub	Urb	Guad	Deg	High	High	No	OffCmps	B5	High	Ave	Yes	Low	High	Ave	Ave	HighLow	HL		A
UG/010/0025	Pub	Rur	Guad	Deg	Med	Med	Yes	OffCmps	B5	Ave	High	Ns	Low	Ave	Low	Ave	HighLow	HL		A
UG/010/0026	Pub	Rur	Par	NoEdu	Med	Med	No	Cmps	5A	Low	Low	Ns	High	Ave	Low	Low	LowLow	LL	A	
UG/010/0027	Pri	Urb	Guad	Deg	Med	High	No	Cmps	B5	High	High	Ns	High	High	High	Ave	HighLow	HL		A
UG/010/0028	Pub	Rur	Par	NoDeg	Med	High	Yes	OffCmps	B5	Ave	Ave	Ns	Low	Ave	High	Ave	LowLow	LL	A	
UG/010/0029	Pub	Rur	Self	NoDeg	Low	Low	Yes	OffCmps	5A	Ave	Ave	Ns	Low	Low	Low	Ave	LowLow	LL	A	

Fig 5.12: Cross-section of the test dataset showing student records, actual result and predicted result obtained from the Prediction Application

For each of the 705 records, the research inputted the 16 features into the predictive application to obtain a calculation of the prediction. The predicted value is inserted in the column called Predicted. Two columns called LowLow and HighLow was created to compare the actual and predicted values. In the columns two values are obtained, the value ‘A’ represents agree and ‘D’ represents disagree. The research produced four count for easy construction of the confusion matrix. In the LowLow column, the A values represents all the students with LowLow that are identified correctly; the D values represent all the students that are incorrectly identified as LowLow. For the HighLow column, the A values represent all the students with HighLow that are identified correctly; the D values represents all the students that are incorrectly identified as HighLow.

Table 5.22: Confusion matrix to discern the accuracy of the predictive application on the test

	Actual LL	Actual HL
Predicted LL	446	11
Predicted HL	3	245

dataset

The results from the table show that 446 students were correctly identified as lowlow students and 245 were correctly identified as highlow students. For the misclassification, 3 students in lowlow group were incorrectly grouped as highlow and 11 students in highlow group were misclassified as lowlow students. From the confusion metrics obtained, this research analyses the predictive application using some metric measures.

Recall or Sensitivity

This is the ratio of correctly identified number of True Positive records from the actual Positive records in the data and it measures the proportion of actual positives correctly identified. The ratio of correctly predicted lowlow students (446) to the actual number of students in the lowlow group (449) is 99.3%. This indicates that the predictive application is 99.3% sensitive in recognizing the students that require high intervention.

Specificity

This is the ratio of correctly identified True Negative records from the actual Negative records in the data and it measures the proportion of negatives correctly identified. The ratio of correctly predicted highlow students (245) to the actual number of students in the highlow group (256) is

95.7%. This indicates that the predictive application is 95.7% specific in recognizing students that require low intervention.

Prevalence

This measures the ratio of the actual positives to the entire dataset. The ratio of the actual lowlow students (449) to the total number of students (705) gives a lowlow prevalence which is 63.7% compared to the ratio of highlow students (256) to the total number of students (705) gives the highlow prevalence which is 36.3%. This shows that the proportion of students requiring high intervention is more than twice the number of students requiring low intervention from the total number of low performing students in NDU.

Accuracy

This is the ratio of correctly identified lowlow and highlow students (691) to the total number of students in the test dataset (705), which is 98%. It measures the performance of the classifier in correctly classifying both groups. This indicates that the predictive application correctly classified 691 students and misclassified 14 students out of the 705 students. The number of misclassification is low and it shows the predictive application has high sensitivity and discrimination power to predict students' classes correctly.

Precision

This is the ratio of correctly predicted True Positives to all the Positive values predicted by the model. The ratio of correctly predicted lowlow students (446) to the total number of students predicted in the lowlow group (457) is 97.6%. Only 11 students out of 256 students in the highlow class was misclassified and placed in the lowlow class. This is not an extremely severe error, as the software achieves a high level of prediction and the misclassified students can benefit from the intervention given.

F-Measure

The F-Measure metric centres on the accuracy of predicting students that require high intervention. This metric is a combination of precision and sensitivity. From the results using the predictive application, 98.5% accuracy was achieved, which means that the predictive application correctly identified 98.5% of students that require high intervention and wrongly classified 1.5% of the students that require high intervention. The software achieves a high level of prediction and the

misclassified students can benefit from the intervention given. The good performance obtained using the model built for testing shows that the model performs very well with the test dataset and can generalise well.

5.4.2 System requirements evaluation

This section evaluates the system to ensure that the software meets all the specified system requirements by looking at the system requirements gathered for this study and outlining the methods used to ensure the software meets the requirements. These requirements considered users of the system by ensuring that the software offers a system that meets the basic needs of its users in an easy and interactive way. Below are the requirements of the system and discussion on methods used in the study.

1. **Simple and interactive interface:** The first requirement is that the software interface must be simple and easy to use. The design of the software met this requirement by offering instructions on the use of the system and providing easy navigation between web pages.
2. **Secured system:** The software requires security that ensures only authorized persons have access to it. The designed system meets this requirement by providing login information of authorized users; the login information comprises username and password stored in an encrypted database.
3. **Prevent prediction errors:** Another requirement of the software is the prevention of prediction errors by ensuring that no student feature field is empty before the prediction process begins. The software met this requirement by ensuring that there is a notification for the user where there is an empty field and all fields are complete before prediction can take place.
4. **Timely and accurate results:** The system also requires timely and accurate results to ensure users get correct predictive results. The software meets this requirement by ensuring the selection of all student feature fields and providing results by one simple click of the “*view result*” button.
5. **Provide risk level:** The final requirement of the system is to ensure that the software predicts the risk level of new students to enable their institution/s to begin providing early intervention. The software meets the requirement by offering a result prediction and summary stating the risk level of the summary and the type of intervention required by the student.

5.5 Chapter Summary

This chapter offered a concise presentation of the last stage of the CRISP-DM process, which is the deployment stage. The knowledge acquired from the modelling and evaluation phases of the methodology led to the deployment stage. For this study, this phase involves using the acquired knowledge to design and implement a predictive system for the classification of low performing undergraduates in NDU into low risk and high risk categories. This categorisation initiates the necessary intervention, almost immediately after the correct prediction. In sum, this chapter evaluated the process of designing and implementing the acquired knowledge in the predictive system using the php programming language connected via the internet to the server.

CHAPTER SIX: SUMMARY AND CONCLUSIONS

6.1 Introduction

The purpose of this study was to build a predictive system that classifies low-performing undergraduate students in NDU into low-risk and high-risk groups by employing the CRISP-DM methodology, which is commonly used in EDM to achieve project goals.

This chapter seeks to confirm whether the goal of this study was achieved by looking at the five research questions outlined in Chapter One and to confirm if they were effectively answered in the study.

1. Which factors are associated with low performance of undergraduates in Nigeria?
2. How could these factors be represented using machine learning techniques?
3. Which machine learning technique can best classify low performing students?
4. What are the best set of features from the total descriptors in the dataset?
5. Can the best machine learning technique and features identified assist in the design of a predictive system to identify low performing students?

The chapter strives to evaluate whether the study answered the questions, how they were answered and how the predictive system developed in this study assists in solving the problem of identifying high-risk low-performing students in NDU. The chapter further makes a case for how this study contributed to new knowledge, the limitations of the study and recommendations for future researchers.

6.2 Evaluation of Research Findings

The study evaluates the research findings specifically by focusing on the efficacy of the answers to all the research questions. In order to answer the research questions, the objectives outlined must be met and the findings from this process aids in assessing if the research accomplished its goal.

6.2.1 Research Question One

The first research question in this study is: “Which factors are associated with low performance of undergraduates in Nigeria?” To answer this question, the research strove to meet the objective below:

To examine and describe factors affecting underperformance of undergraduates in Nigeria by reviewing literature extensively

In Section 2.5.1 the study reviewed the causes of poor academic performance with a focus on Nigeria and some of the causes identified are low financial income, low support from guardians and parents, lack of employment for graduates, poor study plans, family educational background, lack of interest in the course of study, etc. These causes identified from the review assisted the research in identifying relevant features during the data collection process and pre-processing.

6.2.2 Research Question Two

The second research question in this study was “How can these factors be collected and represented in machine-readable format for mining?” To answer this question, the research focused on meeting the objective below:

To collect low performing students’ data in NDU based on factors identified from literature using data capturing techniques and convert the data from source documents to machine readable format using Microsoft Excel

The study in Section 3.3.2.1 described the data collection process by collecting and reviewing available records. These records collected from the university in different formats of pdf and hard copies were converted from the source documents and stored in Microsoft Excel files. The factors, which were represented in the physical files were matched to features from the data collected and represented in the appropriate data mining format prior to mining.

Thus, these converted to machine-readable format assisted the research in the modelling process.

6.2.3 Research Question Three

The third research question in this study is “Which machine learning technique can best classify low performing students?” To answer this question, the research strove to meet the objective below:

To identify the best machine learning technique for classifying low performers in NDU by analysing five machine-learning algorithms were used for classification, namely J48, LR, MLP, NV and SMO

The algorithm that performs the best in classifying the low-performing students' dataset is the multilayer perceptron classifier using values such as Recall, Specificity, ROC, F-Measure, Kappa and RMSE. The study focused on the performance of the five classifiers used in this study in Section 4.2.3 and presents the results from the modelling process using the training dataset to build the classifier models and the test dataset to test the models generated in this study. From the performance of both the training and test dataset and the six metrics of evaluation used, the multilayer perceptron was selected as the best model for the dataset used in this study.

6.2.4 Research Question Four

The fourth research question in this study is “What are the best set of features from the total features in the dataset?” To answer this question, the research attempts to meet the objective below:

To select the best features from the dataset using four feature selection techniques, which are Correlation, Gain Ratio, Information Gain and ReliefF

The best features identified in this study used four feature selection techniques to rank the 24 features using the training dataset and the results in Section 4.3 amplify what this study established. Subsequently, the results from the ranking process assisted in the consecutive modelling of the features ranked; the evaluation of the performance from the successive modelling is presented in Section 4.3.1. From the successive modelling, the study identified the top 16 ranked attributes using the Information Gain algorithm as the best features for the dataset in this study. The 16 features identified are: *Sponsor qualification, Weekly study time, Average SSCE score, Sponsor type, Jamb score, Sponsor support, Secondary school type, Work and study, Sponsor income, Secondary school area, University accommodation, Postgraduate degree, Years before admission, Sports activeness, Post-UTME score and Course interest.*

These 16 features identified were ultimately used in the design of the predictive software for the Niger Delta University.

6.2.5 Research Question Five

The fifth research question in this study is “Can the best machine learning technique and best features identified assist in the design of a predictive system to identify low performing students?” To answer this question, the research strove to meet the objective below:

To use the best machine learning algorithm and the best features identified to design a predictive system for identifying low performers in NDU using PHP programming language

The best features and the best machine learning algorithm identified in this study were used in the design of the predictive system to classify low-performing students in NDU using the PHP programming language. The design of the predictive software was presented in Section 5.4.

6.3 Summary of conclusions

Low academic performance is a challenge for every institution in society and this severely affects the goals of these educational institutions, which is to prepare their scholars for the society by providing quality education that ultimately allows them compete favourably in the society (Velazquez et al, 2006). This low academic performance challenge also affects institutions as universities that record high rate of poor academic performance receive low university rankings on global scales (Dill & Soo, 2005). Furthermore, tertiary institutions regularly come up with policies to enhance their growth, thus they are constantly looking for effective and efficient methods that could create improved policies for their institutions. As stated earlier, low academic performance cuts across every society; however, the challenge is more prominent in developing countries, which has low-income earners, poor access to good medical care, poor electricity and poor funding that only complicate the performance capacities in their intakes (Walker et al, 2007).

Research in recent times has used data mining techniques to gain knowledge about students and their learning patterns, yet scholars have not successfully designed robust and informed models for developing countries (Vahdat et al, 2015). Although some good models exist for scholars in developed countries, it is necessary to design models for developing nations, as the attributes of low performance often vary with the specific contextual factors in every society. Using data mining methods, organizations gain previously unknown knowledge from huge sets of data (Milovic & Milovic, 2012) and since educational institutions regularly produce huge amount of data, this fits quite well. Hence, this research interrogates the possibilities and practicalities of employing machine-learning methods to classify students with low academic performance in a Nigeria as a developing country.

To achieve this goal, this research follows the method of identifying key attributes of low academic performance in Nigeria, comparing the performance of five different machine-learning algorithms, selecting the best features from the entire attributes collected, selecting the best classifier model and developing a predictive software using the best classifier model identified. This proposed

software provides the university with timely and accurate information to identify low performers and assist the university intervene early enough. This research utilises data collected from the Niger Delta University, a public university in Bayelsa state, Nigeria, to achieve the objectives of the research. The development of the predictive system is the most novel contribution of this thesis to the body of knowledge and serves as a platform to solve the problem associated with identifying learners that perform poorly in higher education for developing countries.

6.4 Challenges and Limitations of this Study

This study focused on the classification of low-performing students in only one university in Nigeria. The purpose was to classify low performing students in the university into two groups based on their exposure risk to failure so that new students with high risks of failing can be identified by stakeholders early enough, enabling them provide appropriate plans for intervention. This study also evaluated the software with only the test dataset; therefore, the study undertook no evaluation with new students in the current study.

There were some challenges faced in the data collection process and the subsequent section outlines these challenges.

6.4.1 Data Collection Challenges

Data collection from online data repositories offers extensive data that can be easily located and used (Siemens & Baker, 2012). However, the process of physically collecting stored data from any organization is seriously challenging and requires a lot of time. The challenges experienced during the data collection process are as follows:

1. Delays: the amount of time spent in gathering data; from the time spent on waiting for feedback from stakeholders with regards commencing the actual data collection, the time spent on collecting the data, to the time spent on collating the data proved challenging.
2. Locations of data: data was not available in one physical location and the researcher had to move from one faculty to another. Even at the faculty level, student details are kept by different faculty officers while student performance results are kept by either the heads of departments or departmental examination officers.
3. Missing/incomplete fields: the data contained a huge amount of incomplete/missing fields in the data.

4. Data collation and cleaning: data collation and cleaning was tedious and required a great deal of attention. The researcher verified records to ensure these complied with parameters designed for this study. Each record was double-checked in order to confirm both the value and reliability of such data record.
5. Eliminating bias: the data collected was from only one university in Nigeria. This introduces some form of bias, as the population of students in the university consists mainly of students from the region where the university is located. The aim of this study stated Nigerian higher educational institutions as the population for the study, thus, to eliminate bias this study suggests the use of the predictive application in universities across other regions the country to discover discrepancies and extend the software to fit different regions where necessary.

The challenges experienced during the research were strictly outside the control of the researcher, as the challenges required patience, time and a good attitude towards supporting staff. In situations where the researcher could make adjustments to save time like communicating with two or three faculties concurrently, the researcher wasted no time in doing so.

6.5 Further Research

This study recommends further research in extending the capabilities of the system to include monitoring students' learning patterns, predicting individual students' grades and suggesting intervention methods applicable to stakeholders.

The study focused on only one university in Nigeria, therefore further researchers could extend this use of the software to other higher institutions in developing countries, Nigeria in particular, to enable the development of a unified model that all higher degree institutions can use in the country.

REFERENCES

- Acharya, A. and Sinha, D., 2014. Application of feature selection methods in educational data mining. *International Journal of Computer Applications*, 103(2).
- Adeyemi, A.M. and Adeyemi, S.B., 2014. Institutional Factors as Predictors of Students' Academic Achievement in Colleges of Education in South Western Nigeria. *International Journal of Educational Administration and Policy Studies*, 6(8), pp.141-153.
- Adeyemi, A.M. and Adeyemi, S.B., 2014. Personal factors as predictors of students' academic achievement in colleges of education in South Western Nigeria. *Educational Research and Reviews*, 9(4), pp.97-109.
- Adeyemo, A.B. and Kuye, G., 2006. Mining students' academic performance using decision tree algorithms. *Journal of Information Technology Impact*, 6(3), pp.161-170.
- Adeyemo, O.O., Adeyeye, T.O. and Ogunbiyi, D., 2015. Comparative study of ID3/C4.5 decision tree and multilayer perceptron algorithms for the prediction of typhoid fever. *African Journal of Computing & ICT*, 8(1), pp.103-112.
- Agaoglu, M., 2016. Predicting Instructor Performance Using Data Mining Techniques in Higher Education. *IEEE Access*, 4, pp.2379-2387.
- Aggarwal, C.C., 2015. *Data mining: The textbook*. Springer.
- Ahmed, A.M., Rizaner, A. and Ulusoy, A.H., 2016. Using data mining to predict instructor performance. *Procedia Computer Science*, 102, pp.137-142.
- Ajaja, P.O., 2012. School dropout patterns among senior secondary schools in Delta State, Nigeria. *International Education Studies*, 5(2), p.145.
- Alami, M., 2016. Causes of Poor Academic Performance among Omani Students. *International Journal of Social Science Research*, 4(1), pp.126-136.
- Algarni, A., 2016. Data mining in education. *International Journal of Advanced Computer Science and Applications*, 7(6), pp.456-461.

- Aloise-Young, P.A. and Chavez, E.L., 2002. Not all school dropouts are the same: Ethnic differences in the relation between reason for leaving school and adolescent substance use. *Psychology in the Schools*, 39(5), pp.539-547.
- Al-Radaideh, Q.A. and Al Nagi, E., 2012. Using data mining techniques to build a classification model for predicting employees' performance. *International Journal of Advanced Computer Science and Applications*, 3(2).
- Al-Saleem, M., Al-Kathiry, N., Al-Osimi, S. and Badr, G., 2015, July. Mining educational data to predict students' academic performance. In *International Workshop on Machine Learning and Data Mining in Pattern Recognition* (pp. 403-414). Springer, Cham.
- AlShammari, I., Aldhafiri, M. and Al-Shammari, Z., 2013. A meta-analysis of educational data mining on improvements in learning outcomes. *College Student Journal*, 47(2), pp.326-333.
- Al-Zoubi, S.M. and Younes, M.A.B., 2015. Low academic achievement: Causes and results. *Theory and Practice in Language Studies*, 5(11), pp.2262-2268.
- Amrieh, E.A., Hamtini, T. and Aljarah, I., 2016. Mining educational data to predict students' academic performance using ensemble methods. *International Journal of Database Theory and Application*, 9(8), pp.119-136.
- Anguita, D., Ghio, A., Ridella, S. and Sterpi, D., 2009. K-Fold Cross Validation for Error Rate Estimate in Support Vector Machines. In *DMIN* (pp. 291-297).
- Ani, E.I., 2017. Debating the Roots of Poor Academic Performance in the West African Sub-region: The Perspective of a Philosopher. *SAGE Open*, 7(2), p.2158244017707795.
- Arlot, S. and Celisse, A., 2010. A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4, pp.40-79.
- Arndt, C. and Brefeld, U., 2016. Predicting the future performance of soccer players. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 9(5), pp.373-382.
- Aruna, S. and Rajagopalan, S.P., 2011. A novel SVM based CSSFFS feature selection algorithm for detecting breast cancer. *International Journal of Computer Applications*, 31(8), pp.14-20.

Aryana, M., 2010. Relationship between self-esteem and academic achievement amongst pre-university students. *Journal of Applied Sciences*, 10(20), pp.2474-2477.

Asif, R., Hina, S. and Haque, S.I., 2017. Predicting Student Academic Performance using Data Mining Methods. *IJCSNS*, 17(5), p.187.

Asif, R., Merceron, A. and Pathan, M.K., 2014. Predicting student academic performance at degree level: A case study. *International Journal of Intelligent Systems and Applications*, 7(1), p.49.

Aziz, A.A., Ismail, N.H. and Ahmad, F., 2013. Mining Students' academic Performance. *Journal of Theoretical & Applied Information Technology*, 53(3), 485-495.

Babalola, J.B., 2015. Achieving Nigerian Developmental and Educational Goals through Incorporation of Best Practices in National Education Policies: Priority interventions, proven practices and policy problems: A lead paper presented in the University of Nigeria. Nsukka, Nigeria.

Badr, G., Algobail, A., Almutairi, H. and Almutery, M., 2016. Predicting students' performance in university courses: a case study and tool in KSU mathematics department. *Procedia Computer Science*, 82, pp.80-89.

Baker, R., Walonoski, J., Heffernan, N., Roll, I., Corbett, A. and Koedinger, K., 2008. Why students engage in "gaming the system" behavior in interactive learning environments. *Journal of Interactive Learning Research*, 19(2), pp.185-224.

Baker, R.S. and Inventado, P.S., 2014. Educational data mining and learning analytics. In *Learning analytics* (pp. 61-75). Springer, New York, NY.

Baker, R.S. and Yacef, K., 2009. The state of educational data mining in 2009: A review and future visions. *JEDM/ Journal of Educational Data Mining*, 1(1), pp.3-17.

Baker, R.S., 2007. Modeling and understanding students' off-task behavior in intelligent tutoring systems. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 1059-1068). ACM.

Baker, R.S., 2010. Data mining for education. *International Encyclopaedia of Education*, 7(3), pp.112-118.

- Baker, R.S., 2010. Mining data for student models. In *Advances in Intelligent Tutoring Systems* (pp. 323-337). Springer, Berlin, Heidelberg.
- Baker, R.S., Corbett, A.T. and Koedinger, K.R., 2004. Detecting student misuse of intelligent tutoring systems. In *International Conference on Intelligent Tutoring Systems* (pp. 531-540). Springer, Berlin, Heidelberg.
- Bakhshinategh, B., Zaiane, O.R., ElAtia, S. and Ipperciel, D., 2018. Educational data mining applications and tasks: A survey of the last 10 years. *Education and Information Technologies*, 23(1), pp.537-553.
- Baldi, P., Brunak, S., Chauvin, Y., Andersen, C.A. and Nielsen, H., 2000. Assessing the accuracy of prediction algorithms for classification: An overview. *Bioinformatics*, 16(5), pp.412-424.
- Ballard, C., Rollins, J., Ramos, J., Perkins, A., Hale, R., Doerneich, A., Milner, E.C. and Chodagam, J., 2007. *Dynamic Warehousing: Data Mining Made Easy*. IBM Redbooks Series.
- Banerjee, P.A., 2016. A systematic review of factors linked to poor academic performance of disadvantaged students in science and maths in schools. *Cogent Education*, 3(1), p.1178441.
- Baradwaj, B.K. and Pal, S., 2012. Mining educational data to analyse students' performance. *arXiv preprint arXiv:1201.3417*.
- Beniwal, S. and Arora, J., 2012. Classification and feature selection techniques in data mining. *International Journal of Engineering Research & Technology (IJERT)*, 1(6), pp. 1-6.
- Berendt, B., Littlejohn, A., Kern, P., Mitros, P., Shacklock, X. and Blakemore, M., 2017. *Big Data for monitoring educational systems*. Publications Office of the European Union, Luxembourg.
- Berkowitz, R., Moore, H., Astor, R.A. and Benbenishty, R., 2017. A research synthesis of the associations between socioeconomic background, inequality, school climate, and academic achievement. *Review of Educational Research*, 87(2), pp.425-469.
- Bienkowski, M., Feng, M. and Means, B., 2012. Enhancing teaching and learning through educational data mining and learning analytics: An issue brief. *US Department of Education, Office of Educational Technology*, 1, pp.1-57.

- Bolapeju, M., Adeyemi, J.K. and Ogbodo, C.M., 2014. Academic achievement and admission policy as correlate of student retention in Nigerian federal universities. *International Journal of Business and Social Science*, 5(2).
- Borkar, S. and Rajeswari, K., 2014. Attributes selection for predicting students' academic performance using education data mining and artificial neural network. *International Journal of Computer Applications*, 86(10).
- Botchkarev, A., 2018. Performance Metrics (Error Measures) in Machine Learning Regression, Forecasting and Prognostics: Properties and Typology. *arXiv preprint arXiv:1809.03006*.
- Boughorbel, S., Jarray, F. and El-Anbari, M., 2017. Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric. *PlosOne*, 12(6), p.e0177678.
- Bucos, M. and Drăgulescu, B., 2018. Predicting Student Success Using Data Generated in Traditional Educational Environments. *TEM Journal – Technology Education Management Informatics*, 7(3), pp.617-625.
- Caruana, R. and Niculescu-Mizil, A., 2004, August. Data mining in metric space: An empirical analysis of supervised learning performance criteria. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 69-78). ACM.
- Cen, H., Koedinger, K. and Junker, B., 2006, June. Learning factors analysis—a general method for cognitive model evaluation and improvement. In *International Conference on Intelligent Tutoring Systems* (pp. 164-175). Springer, Berlin, Heidelberg.
- Chai, T. and Draxler, R.R., 2014. Root mean square error (RMSE) or mean absolute error (MAE)? Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, 7(3), pp.1247-1250.
- Chalaris, M., Gritzalis, S., Maragoudakis, M., Sgouropoulou, C. and Tsolakidis, A., 2014. Improving quality of educational processes providing new knowledge using data mining techniques. *Procedia-Social and Behavioral Sciences*, 147, pp.390-397.
- Chandrashekar, G. and Sahin, F., 2014. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1), pp.16-28.

- Chang, H.W., Chiu, Y.H., Kao, H.Y., Yang, C.H. and Ho, W.H., 2013. Comparison of classification algorithms with wrapper-based feature selection for predicting osteoporosis outcome based on genetic factors in a Taiwanese women population. *International Journal of Endocrinology*, p.850735.
- Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R., 2000. *CRISP-DM 1.0 Step-by-step data mining guide*. SPSS Inc., USA.
- Chiş, M., 2008. Evolutionary decision trees and software metrics for module defects identification. *Proceedings of World Academy of Science, Engineering and Technology*, 28, pp.273-277.
- Das, S., 2001, June. Filters, wrappers and a boosting-based hybrid for feature selection. In *ICML* (Vol. 1, pp. 74-81).
- Dash, M. and Liu, H., 1997. Feature selection for classification. *Intelligent Data Analysis*, 1(1-4), pp.131-156.
- Daud, A., Aljohani, N.R., Abbasi, R.A., Lytras, M.D., Abbas, F. and Alowibdi, J.S., 2017, April. Predicting student performance using advanced learning analytics. In *Proceedings of the 26th International Conference on World Wide Web Companion* (pp. 415-421). International World Wide Web Conferences Steering Committee.
- David, L. and Gómez, C.E.P., 2014. Contributions from Data Mining to Study Academic Performance of Students of a Tertiary Institute. *American Journal of Educational Research*, 2(9), pp.713-726.
- De Marchi, L., 2011. Data mining of sports performance data. Doctoral dissertation, University of Leeds, School of Computing Studies.
- Dekker, G.W., Pechenizkiy, M. and Vleeshouwers, J.M., 2009. Predicting Students Drop Out: A Case Study. *International Working Group on Educational Data Mining*.
- Desmarais, M., 2011. Conditions for effectively deriving a q-matrix from data with non-negative matrix factorization. In *4th international conference on educational data mining, EDM* (pp. 41-50).

Devadiga, N.M., 2017, October. Tailoring architecture centric design method with rapid prototyping. In *2017 second International Conference on Communication and Electronics Systems (ICCES)* (pp. 924-930). IEEE.

Doll, J.J., Eslami, Z. and Walters, L., 2013. Understanding why students drop out of high school, according to their own reports: Are they pushed or pulled, or do they fall out? A comparative analysis of seven nationally representative studies. *Sage Open*, 3(4), p.2158244013503834.

ElGamal, A.F., 2013. An educational data-mining model for predicting student performance in programming course. *International Journal of Computer Applications*, 70(17), pp.22-28.

Eno-Abasi Sunday, Gbenga Salau, Kelvin Ebiri, Lawrence Njoku, Saxone Akhaine, Murtala Adewale and Charles Ogugbuaja 2018. *Nigerian students and vanishing scholarships, bursaries*. Accessed 28 November 2018, <https://guardian.ng/saturday-magazine/cover/nigerian-students-and-vanishing-scholarships-bursaries/>

Fancsali, S., 2012. Variable construction and causal discovery for cognitive tutor log data: Initial results. In *Educational Data Mining 2012* (pp.238-239).

García, E.P.I. and Mora, P.M., 2011, November. Model prediction of academic performance for first year students. In *Artificial Intelligence (MICAI), 2011 10th Mexican International Conference on* (pp. 169-174). IEEE.

Gayatri, N., Nickolas, S., Reddy, A.V. and Chitra, R., 2009, October. Performance analysis of datamining algorithms for software quality prediction. In *Advances in Recent Technologies in Communication and Computing, 2009. ARTCom'09. International Conference on* (pp. 393-395). IEEE.

Glewwe, P. and Kremer, M., 2006. Schools, teachers, and education outcomes in developing countries. *Handbook of the Economics of Education*, 2, pp.945-1017.

Guo, H., Yin, J., Zhao, J., Yao, L., Xia, X. and Luo, H., 2015. An ensemble learning for predicting breakdown field strength of polyimide nano-composite films. *Journal of Nano-materials*, 2015, p.7.

Guri-Rosenblit, S., 2006. Eight paradoxes in the implementation process of e-learning in higher education. *Distances et savoirs*, 4(2), pp.155-179.

- Hämäläinen, W. and Vinni, M., 2010. Classifiers for educational data mining. *Handbook of Educational Data Mining*, pp.57-74.
- Hameed, S.S., Petinrin, O.O., Osman, A. and Hashi, F.S., 2018. Filter-Wrapper Combination and Embedded Feature Selection for Gene Expression Data. *Int. J. of Advanced Soft Compu. Appl*, 10(1).
- Han, J., Pei, J. and Kamber, M., 2011. *Data Mining: Concepts and Techniques*. Elsevier.
- Henke, N., Bughin, J., Chui, M., Manyika, J., Saleh, T., Wiseman, B. and Sethupathy, G., 2016. The age of analytics: Competing in a data-driven world. *McKinsey Global Institute*, 4.
- Hira, Z.M. and Gillies, D.F., 2015. A review of feature selection and feature extraction methods applied on microarray data. *Advances in Bioinformatics*, 2015.
- Hripcsak, G., 2012. Visualizing the operating range of a classification system. *Journal of the American Medical Informatics Association*, 19(4), pp.529-532.
- Hsu, C.W. and Lin, C.J., 2002. A comparison of methods for multi-class support vector machines. *IEEE transactions on Neural Networks*, 13(2), pp.415-425.
- Husmeier, D., 2005. Introduction to learning Bayesian networks from data. In *Probabilistic modeling in bioinformatics and medical informatics* (pp. 17-57). Springer, London.
- Hussain, S., Dahan, N.A., Ba-Alwib, F.M. and Ribata, N., 2018. Educational data mining and analysis of students' academic performance using WEKA. *Indonesian Journal of Electrical Engineering and Computer Science*, 9(2), pp.447-459.
- Jantan, H., Hamdan, A.R. and Othman, Z.A., 2009. Knowledge discovery techniques for talent forecasting in human resource application. *World Academy of Science, Engineering and Technology*, 50, pp.775-783.
- Kabakchieva, D., 2013. Predicting student performance by using data mining methods for classification. *Cybernetics and Information Technologies*, 13(1), pp.61-72.
- Kantardzic, M., 2011. *Data mining: Concepts, models, methods, and algorithms*. John Wiley & Sons.

Karpagavadivu, K., Maragatham, T. and Karthik, S., 2012. A Survey of Different Software Fault Prediction Using Data Mining Techniques Methods. *International Journal of Advanced Research in Computer Engineering & Technology*, 1(8), pp.1-3.

Kasemsap, K., 2015. The role of data mining for business intelligence in knowledge management. In *Integration of data mining in business intelligence systems* (pp. 12-33). IGI Global.

Kassarnig, V., Mones, E., Bjerre-Nielsen, A., Sapiezynski, P., Lassen, D.D. and Lehmann, S., 2018. Academic performance and behavioral patterns. *EPJ Data Science*, 7(1), p.10.

Katamei, J.M. and Omwono, G.A., 2015. Intervention Strategies to Improve Students' Academic Performance in Public Secondary Schools in Arid and Semi-Arid Lands in Kenya. *Int'l J. Soc. Sci. Stud.*, 3, pp.107-120.

Kaur, R. and Sharma, E.S., 2018. Various Techniques to Detect and Predict Faults in Software System: Survey. *International Journal on Future Revolution in Computer Science & Communication Engineering*, 4(2), pp.330-336.

Kim, S.W., Cho, H. and Kim, L.Y., 2019. Socioeconomic Status and Academic Outcomes in Developing Countries: A Meta-Analysis. *Review of Educational Research*, 89(6), pp.875-916.

Kirimi, J.M. and Moturi, C.A., 2016. Application of Data Mining Classification in Employee Performance Prediction. *International Journal of Computer Applications (0975-8887)*, 146(7).

Kleinbaum, D.G. and Klein, M., 2010. *Logistic Regression: A Self-Learning Text*. Springer Science & Business Media.

Kordon, F., 2002. An introduction to rapid system prototyping. *IEEE Transactions on Software Engineering*, 28(9), pp.817-821.

Kruse, R., Borgelt, C., Braune, C., Mostaghim, S. and Steinbrecher, M., 2016. *Computational intelligence: A methodological introduction*. Springer.

Kumar, G. and Bhatia, P.K., 2014, February. Comparative analysis of software engineering models from traditional to modern methodologies. In *2014 Fourth International Conference on Advanced Computing & Communication Technologies* (pp. 189-196). IEEE.

- Kumar, P. and Wahid, A., 2016. Performance Evaluation of Data Mining Techniques for Predicting Software Reliability. *World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering*, 9(8), pp.2041-2048.
- Larose, D.T. and Larose, C.D., 2014. *Discovering knowledge in data: An introduction to data mining*. John Wiley & Sons.
- Latif, A., Choudhary, A.I. and Hammayun, A.A., 2015. Economic effects of student dropouts: A comparative study. *Journal of Global Economics*.
- Lever, J., Krzywinski, M. and Altman, N.S., 2016. Points of Significance: Model selection and overfitting. *Nature Methods*, 13(9), pp.703-704.
- Liñán, L.C. and Pérez, Á.A.J., 2015. Educational Data Mining and Learning Analytics: differences, similarities, and time evolution. *International Journal of Educational Technology in Higher Education*, 12(3), pp.98-112.
- Longe, O., 2017. Graduate Unemployment in Nigeria: Causes, Consequences and Remediable Approaches. *American International Journal of Contemporary Research*, 7(4), pp.63-73.
- Luna, D.R., Mayan, J.C., García, M.J., Almerares, A.A. and Househ, M., 2014. Challenges and potential solutions for big data implementations in developing countries. *Yearbook of Medical Informatics*, 23(01), pp.36-41.
- Luna, J.M., Castro, C. and Romero, C., 2017. MDM tool: A data mining framework integrated into Moodle. *Computer Applications in Engineering Education*, 25(1), pp.90-102.
- Macfadyen, L.P. and Sorenson, P., 2010, June. Using LiMS (the learner interaction monitoring system) to track online learner engagement and evaluate course design. In *Educational Data Mining 2010*.
- Mardikyan, S. and Badur, B., 2011. Analyzing Teaching Performance of Instructors Using Data Mining Techniques. *Informatics in Education*, 10(2), pp.245-257.
- Márquez-Vera, C., Cano, A., Romero, C. and Ventura, S., 2013. Predicting student failure at school using genetic programming and different data mining approaches with high dimensional and imbalanced data. *Applied intelligence*, 38(3), pp.315-330.

- Márquez-Vera, C., Morales, C.R. and Soto, S.V., 2013. Predicting school failure and dropout by using data mining techniques. *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*, 8(1), pp.7-14.
- Marsland, S., 2014. *Machine learning: an algorithmic perspective*. Chapman and Hall/CRC.
- McHugh, M.L., 2012. Interrater reliability: the kappa statistic. *Biochemia medica*, 22(3), pp.276-282.
- Meenakumari, J. and Kudari, J.M., 2015. Learning Analytics and its challenges in Education Sector a Survey. *International Conference on Current Trends in Advanced Computing (ICCTAC-2015)*, pp.6-10.
- Milovic, B. and Milovic, M., 2012. Prediction and decision making in health care using data mining. *Kuwait Chapter of the Arabian Journal of Business and Management Review*, 1(12), p.126.
- Mirashrafi, S.B., 2013. The Effect of Family Background and Socioeconomic Status on Academic Performance of Higher Education Applicants. *International Journal of Technology and Inclusive Education*, 2(1), pp.130-137.
- Mitra, D., 2011. The social and economic benefits of public education. *Pennsylvania, State University, USA*.
- Mueen, A., Zafar, B. and Manzoor, U., 2016. Modeling and predicting students' academic performance using data mining techniques. *International Journal of Modern Education and Computer Science*, 8(11), p.36.
- Muralidharan, K., 2017. Field experiments in education in developing countries. In *Handbook of economic field experiments* (Vol. 2, pp. 323-385). North-Holland.
- Mushtaq, I. and Khan, S.N., 2012. Factors Affecting Students Academic Performance. *Global Journal of Management and Business Research*, 12(9).
- Mwadulo, M.W., 2016. A review on feature selection methods for classification tasks. *International Journal of Computer Applications Technology and Research*, 5(6), pp.395-402.

National Universities Commission (NUC), no date. *Nigerian Universities*. Accessed 21 September 2018. <http://nuc.edu.ng/nigerian-universities/>.

Neumann, U., Riemenschneider, M., Sowa, J.P., Baars, T., Kälsch, J., Canbay, A. and Heider, D., 2016. Compensation of feature selection biases accompanied with improved predictive performance for binary classification by using a novel ensemble feature selection approach. *BioData Mining*, 9(1), p.36.

Niger Delta University (NDU), no date. *NDU Profile & Environment*. Accessed 11 September 2017, <http://www.ndu.edu.ng/nduprofile.html#>

Nisbet, R., Miner, G. and Yale, K., 2017. *Handbook of Statistical Analysis and Data Mining Applications*. Elsevier.

Nnamani, C.N., Dikko, H.G. and Kinta, L.M., 2014. Impact of students' financial strength on their academic performance: Kaduna Polytechnic experience. *African Research Review*, 8(1), pp.83-98.

Nsiah, H., 2017. Fear of Failure and the Academic Performance of Students from Low-Income Families. *International Journal of Education and Social Science*, 4(10), pp. 19-26.

Nurmi, J.E., Aunola, K., Salmela-Aro, K. and Lindroos, M., 2003. The role of success expectation and task-avoidance in academic performance and satisfaction: Three studies on antecedents, consequences and correlates. *Contemporary educational psychology*, 28(1), pp.59-90.

Ocuppaugh, J., Baker, R., Gowda, S., Heffernan, N. and Heffernan, C., 2014. Population validity for Educational Data Mining models: A case study in affect detection. *British Journal of Educational Technology*, 45(3), pp.487-501.

Ogor, E.N., 2007, September. Student academic performance monitoring and evaluation using data mining techniques. In *Electronics, Robotics and Automotive Mechanics Conference, 2007. CERMA 2007* (pp. 354-359). IEEE.

Okubanjo, A.O., 2008. Parent's Education, Job Type and Occupational Status as Determinants of Fear of Unemployment among Nigerian University Undergraduates. *Ogun Journal of Counseling Studies* 2(2), pp. 141-147.

- Ola, A. and Pallaniappan, S., 2013. A data mining model for evaluation of instructors' performance in higher institutions of learning using machine learning algorithms. *International Journal of Conceptions on Computing and Information Technology*, 1(1), pp. 17-22.
- Olcay, G.A. and Bulu, M., 2017. Is measuring the knowledge creation of universities possible?: A review of university rankings. *Technological Forecasting and Social Change*, 123, pp.153-160.
- Ololube, N.P., 2013. The problems and approaches to educational planning in Nigeria: A theoretical observation. *Mediterranean Journal of Social Sciences*, 4(12), pp.37-48.
- Olotu, A., Salami, R. and Akeremale, I., 2015. Poverty and rate of unemployment in Nigeria. *IJM* 2(1), pp. 1-4.
- Olson, D.L. and Delen, D., 2008. *Advanced data mining techniques*. Springer Science & Business Media.
- Oreski, D., Pihir, I. and Konecki, M., 2017. CRISP-DM Process Model in Educational Setting. *Economic and Social Development: Book of Proceedings*, pp.19-28.
- Osmanbegovic, E. and Suljic, M., 2012. Data mining approach for predicting student performance. *Economic Review: Journal of Economics and Business*, 10(1), pp.3-12.
- Oyebade, S.A. and Dike, C., 2013. Restructuring Nigerian Tertiary (University) Education for Better Performance. *Bulgarian Comparative Education Society*.
- Oyerinde, O.D. and Chia, P.A., 2017. Predicting students' academic performances - A learning analytics approach using multiple linear regression. *International Journal of Computer Applications*, 157(4), pp.37-44.
- Panchal, G., Ganatra, A., Kosta, Y.P. and Panchal, D., 2011. Behaviour analysis of multilayer perceptrons with multiple hidden neurons and hidden layers. *International Journal of Computer Theory and Engineering*, 3(2), pp.332-337.
- Paramshetti, P. and Phalke, D.A., 2014. Survey on software defect prediction using machine-learning techniques. *International Journal of Science and Research (IJSR)*, 3(12), pp.1394-1397.
- Pardos, Z.A., Wang, Q.Y. and Trivedi, S., 2012. The Real World Significance of Performance Prediction. *International Educational Data Mining Society*.

- Park, H., 2013. An introduction to logistic regression: from basic concepts to interpretation with particular attention to nursing domain. *Journal of Korean Academy of Nursing*, 43(2), pp.154-164.
- Peña-Ayala, A., 2014. Educational data mining: A survey and a data mining-based analysis of recent works. *Expert Systems with Applications*, 41(4), pp.1432-1462.
- Pitt, E. and Nayak, R., 2007, December. The use of various data mining and feature selection methods in the analysis of a population survey dataset. In *Proceedings of the 2nd International Workshop on Integrating Artificial Intelligence and Data Mining Volume 84* (pp. 83-93). Australian Computer Society, Inc.
- Platt, J., 1998. Sequential minimal optimization: A fast algorithm for training support vector machines. *Microsoft Research, Technical Report MSR-TR-98-14*, pp. 1-21.
- Rahm, E. and Do, H.H., 2000. Data cleaning: Problems and current approaches. *IEEE Data Eng. Bull.*, 23(4), pp.3-13
- Rai, D., and Beck, J. 2011. Exploring user data from a game-like math tutor: A case study in causal modeling. In *4th International Conference on Educational Data Mining* (pp. 307–313).
- Ramaswami, M. and Bhaskaran, R., 2009. A study on feature selection techniques in educational data mining. *arXiv preprint arXiv:0912.3924*.
- Raosoft, 2004. Sample Size Calculator. Available at: <http://www.raosoft.com/samplesize.html>.
- Rodrigo, M., Mercedes, T., d Baker, R.S., McLaren, B.M., Jayme, A. and Dy, T.T., 2012. Development of a Workbench to Address the Educational Data Mining Bottleneck. In *5th International Conference on Educational Data Mining*, pp. 152-155
- Romero, C. and Ventura, S., 2007. Educational data mining: A survey from 1995 to 2005. *Expert Systems with Applications*, 33(1), pp.135-146.
- Romero, C. and Ventura, S., 2020. Educational data mining and learning analytics: An updated survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10(3), p.e1355.
- Romero, C. and Ventura, S., 2010. Educational data mining: a review of the state of the art. *IEEE Transactions on Systems, Man, and Cybernetics*, 40(6), pp.601-618.

- Romero, C. and Ventura, S., 2013. Data mining in education. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 3(1), pp.12-27.
- Romero, C., Ventura, S., Espejo, P.G. and Hervás, C., 2008. Data mining algorithms to classify students. In *1st International Conference on Educational Data Mining*, pp.8-17
- Romero, C., Ventura, S., Pechenizkiy, M. and Baker, R.S. eds., 2010. *Handbook of educational data mining*. CRC press.
- Rountree, N., Rountree, J., Robins, A. and Hannah, R., 2004. Interacting factors that predict success and failure in a CS1 course. In *ACM SIGCSE Bulletin*, 36(4), pp. 101-104.
- Rovira, S., Puertas, E. and Igual, L., 2017. Data-driven system to predict academic grades and dropout. *PlosOne*, 12(2), p.e0171207.
- Sabourin, J., Kosturko, L., FitzGerald, C. and McQuiggan, S., 2015. Student Privacy and Educational Data Mining: Perspectives from Industry. *International Educational Data Mining Society*.
- Sachin, R.B. and Vijay, M.S., 2012, January. A survey and future vision of data mining in educational field. In *Advanced Computing & Communication Technologies (ACCT), 2012 Second International Conference on* (pp. 96-100). IEEE
- Saeys, Y., Abeel, T. and Van de Peer, Y., 2008, September. Robust feature selection using ensemble feature selection techniques. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (pp. 313-325). Springer, Berlin, Heidelberg.
- Saeys, Y., Inza, I. and Larrañaga, P., 2007. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19), pp.2507-2517.
- Saito, T. and Rehmsmeier, M., 2015. The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PlosOne*, 10(3), p.e0118432.
- Sánchez-Marroño, N., Alonso-Betanzos, A. and Tombilla-Sanromán, M., 2007, December. Filter methods for feature selection—a comparative study. In *International Conference on Intelligent Data Engineering and Automated Learning* (pp. 178-187). Springer, Berlin, Heidelberg.

Sarlis, N.V. and Christopoulos, S.R.G., 2014. Visualization of the significance of Receiver Operating Characteristics based on confidence ellipses. *Computer Physics Communications*, 185(3), pp.1172-1176.

Sasaki, Y., 2007. The truth of the F-measure. *Teach Tutor Matters*, 1(5), pp.1-5.

Segue Technologies. (2015). *The Benefits of Adhering to a Software Development Methodology*. [Online] Available at: <https://www.seguetech.com/benefits-adhering-software-development-methodology-concepts/> [Accessed 15 Oct. 2019].

Selmoune, N. and Alimazighi, Z., 2008, April. A decisional tool for quality improvement in higher education. In *Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008*. 3rd International Conference, IEEE, pp.1-6.

Sengupta, S. and Bhattacharya, S., 2006, June. Formalization of UML use case diagram-a Z notation based approach. In *2006 International Conference on Computing & Informatics* (pp. 1-6). IEEE.

Shahiri, A.M., Husain, W., Nur'aini, A.R., 2015. A review on predicting student's performance using data mining techniques. *Procedia Computer Science*, 72, pp.414-422.

Shaikh, A., Mahoto, N., Khuhawar, F. and Memon, M., 2015. Performance evaluation of classification methods for heart disease dataset. *Sindh University Research Journal-SURJ (Science Series)*, 47(3).

Shannon, G.S. and Bylsma, P., 2006. Helping Students Finish School: Why Students Drop Out and How to Help Them Graduate. *Washington Office of Superintendent of Public Instruction*.

Shardlow, M., 2016. An analysis of feature selection techniques. *The University of Manchester*, pp.1-7.

Shields, J., Brown, M., Kaine, S., Dolle-Samuel, C., North-Samardzic, A., McLean, P., Johns, R., O'Leary, P., Robinson, J. and Plimmer, G., 2015. *Managing employee performance & reward: Concepts, practices, strategies*. Cambridge University Press.

Shu, Y., Liu, H., Wu, Z. and Yang, X., 2009. Modeling of software fault detection and correction processes based on the correction lag. *Information Technology Journal*, 8(5), pp.735-742.

Siemens, G. and Baker, R.S., 2012, April. Learning analytics and educational data mining: towards communication and collaboration. In *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge* (pp. 252-254). ACM.

Singh, S. and Gupta, P., 2014. Comparative study ID3, cart and C4. 5-decision tree algorithm: a survey. *International Journal of Advanced Information Science and Technology (IJAIST)*, 27(27), pp.97-103.

Smith, M., Szongott, C., Henne, B. and Von Voigt, G., 2012. Big data privacy issues in public social media. In *Digital Ecosystems Technologies (DEST), 2012 6th IEEE International Conference on* (pp. 1-6). IEEE.

Sokolova, M. and Lapalme, G., 2009. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4), pp.427-437.

Srivastava, S.K. and Singh, S.K., 2015. Multi-parameter based performance evaluation of classification algorithms. *Int. J. Comput Sci. Inform. Technol. (IJCSIT)*, 7, pp.115-125.

Stinebrickner, R. and Stinebrickner, T., 2014. Academic performance and college dropout: Using longitudinal expectations data to estimate a learning model. *Journal of Labor Economics*, 32(3), pp.601-644.

Strecht, P., Cruz, L., Soares, C., Mendes-Moreira, J. and Abreu, R., 2015. A Comparative Study of Classification and Regression Algorithms for Modelling Students' Academic Performance. *International Educational Data Mining Society*.

Strohmeier, S. and Piazza, F., 2013. Domain driven data mining in human resource management: A review of current research. *Expert Systems with Applications*, 40(7), pp.2410-2420.

Sukhija, K., Jindal, M. and Aggarwal, N., 2015, October. The recent state of educational data mining: A survey and future visions. In *MOOCs, Innovation and Technology in Education (MITE), 2015 IEEE 3rd International Conference on* (pp. 354-359). IEEE.

Sommerville, I., 2011. *Software engineering*. 9th Edition. ISBN-10, 137035152, Pearson Education, America.

Suthaharan, S., 2016. Support vector machine. In *Machine learning models and algorithms for big data classification* (pp. 207-235). Springer, Boston, MA.

Tam, V., Lam, E.Y., Fung, S.T., Fok, W.W.T. and Yuen, A.H., 2015, December. Enhancing educational data mining techniques on online educational resources with a semi-supervised learning approach. In *Teaching, Assessment, and Learning for Engineering (TALE), 2015 IEEE International Conference on* (pp. 203-206). IEEE.

Tan, P.N., Steinbach, M. and Kumar, V., 2013. *Introduction to Data Mining*. Pearson Education Limited.

The World University Rankings, 2018. *World University Rankings 2019: Methodology*. Accessed 10th November 2018, <https://www.timeshighereducation.com/world-university-rankings/methodology-world-university-rankings-2019>.

Ting K.M., 2017. Confusion Matrix. In: *Sammut C., Webb G.I. (eds) Encyclopaedia of Machine Learning and Data Mining*. Springer, Boston, MA.

Ugar, A.A., 2018. ASUU Strike: The Federal Government and Nigerian Educational System. *International Journal of Education and Research*, 6(5), pp.19-32.

Vahdat, M., Ghio, A., Oneto, L., Anguita, D., Funk, M. and Rauterberg, M., 2015. Advances in learning analytics and educational data mining. *Proc. of ESANN2015*, pp.297-306.

Valle, M.A., Varas, S. and Ruz, G.A., 2012. Job performance prediction in a call center using a naive Bayes classifier. *Expert Systems with Applications*, 39(11), pp.9939-9945.

Vandamme, J.P., Meskens, N. and Superby, J.F., 2007. Predicting academic performance by data mining methods. *Education Economics*, 15(4), pp.405-419.

Vernon, M.M., Balas, E.A. and Momani, S., 2018. Are university rankings useful to improve research? A systematic review. *PloS one*, 13(3).

Wang, W. and Lu, Y., 2018, March. Analysis of the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE) in Assessing Rounding Model. In *IOP Conference Series: Materials Science and Engineering* (Vol. 324, No. 1, p. 012049). IOP Publishing.

WES Staff, 2017. *Education in Nigeria*. Accessed 21 September 2018, <https://wenr.wes.org/2017/03/education-in-nigeria>.

Willmott, C.J. and Matsuura, K., 2005. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate research*, 30(1), pp.79-82.

Winston, K.A., van der Vleuten, C.P. and Scherpbier, A.J., 2014. Prediction and prevention of failure: An early intervention to assist at-risk medical students. *Medical Teacher*, 36(1), pp.25-31.

Wirth, R. and Hipp, J., 2000, April. 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* (pp. 29-39). Citeseer.

Wixon, M., d Baker, R.S., Gobert, J.D., Ocumpaugh, J. and Bachmann, M., 2012, July. WTF? Detecting students who are conducting inquiry without thinking fastidiously. In *International Conference on User Modeling, Adaptation, and Personalization* (pp. 286-296). Springer, Berlin, Heidelberg.

Wood, J.M., 2007. Understanding and Computing Cohen's Kappa: A Tutorial. *WebPsychEmpiricist. Web Journal at <http://wpe.info/>*.

Wu, Y.P., Hu, Q.P. and Ng, S.H., 2006, June. A study of software fault detection and correction process models. In *Management of Innovation and Technology, 2006 IEEE International Conference on* (Vol. 2, pp. 812-816). IEEE.

Yang, D., Sinha, T., Adamson, D. and Rosé, C.P., 2013, December. Turn on, tune in, drop out: Anticipating student dropouts in massive open online courses. In *Proceedings of the 2013 NIPS Data-driven education workshop* (Vol. 11, p. 14).

Yehuala, M.A., 2015. Application of Data Mining Techniques for Student Success and Failure Prediction (The Case of Debre_Markos University). *International Journal of Scientific & Technology Research*, 4(4), pp.91-94.

Yukselturk, E., Ozekes, S. and Türel, Y.K., 2014. Predicting dropout student: An application of data mining methods in an online education program. *European Journal of Open, Distance and E-learning*, 17(1), pp.118-133.

Zaki, M.J., Meira Jr, W. and Meira, W., 2014. *Data mining and analysis: Fundamental concepts and algorithms*. Cambridge University Press.

Zeng, Z.Q., Yu, H.B., Xu, H.R., Xie, Y.Q. and Gao, J., 2008, November. Fast training support vector machines using parallel sequential minimal optimization. In *2008 3rd international conference on intelligent system and knowledge engineering* (Vol. 1, pp. 997-1001). IEEE.

APPENDIX



Private Bag X6001, Potchefstroom,
South Africa, 2520

Tel: (018) 299-4900

Faks: (018) 299-4910

Web: <http://www.nwu.ac.za>

Research Ethics Regulatory Committee

Tel: +27 18 299 4849

Email: Ethics@nwu.ac.za

ETHICAL CLEARANCE LETTER OF STUDY

Based on approval by the **Health Science Ethics Committee (FAST-HSEC)** on **02/10/2018** after being reviewed at the meeting held on **02/10/2018**, the North-West University Research Ethics Regulatory Committee (NWU-RERC) hereby **approves** your project as indicated below. This implies that the NWU-RERC grants its permission that, provided the special conditions specified below are met and pending any other authorisation that may be necessary, the project may be initiated, using the ethics number below.

Project title: Predictive system for characterizing low performance of Undergraduate students using machine learning techniques.																														
Project Leader/Supervisor: Prof BM Esiefarienrhe & Prof N Gasela																														
Student: EA Ekubo																														
Ethics number:	<table border="1"><tr><td>N</td><td>W</td><td>U</td><td>-</td><td>0</td><td>0</td><td>6</td><td>0</td><td>3</td><td>-</td><td>1</td><td>8</td><td>-</td><td>A</td><td>9</td></tr><tr><td colspan="3">Institution</td><td></td><td colspan="5">Year</td><td colspan="3">Status</td><td></td><td></td></tr></table> <small>Status: S = Submission; R = Re-Submission; P = Provisional Authorisation; A = Authorisation</small>	N	W	U	-	0	0	6	0	3	-	1	8	-	A	9	Institution				Year					Status				
N	W	U	-	0	0	6	0	3	-	1	8	-	A	9																
Institution				Year					Status																					
Application Type: Single study																														
Commencement date: 2018-10-02	Expiry date: 2019-10-02																													
Risk:	Minimum																													

Special conditions of the approval (if applicable):

General conditions:

While this ethics approval is subject to all declarations, undertakings and agreements incorporated and signed in the application form, the following general terms and conditions will apply:

- The project leader (principle investigator) must report in the prescribed format to the HSEC:
 - annually (or as otherwise requested) on the progress of the project, and upon completion of the project;
 - without any delay in case of any adverse event (or any matter that interrupts sound ethical principles) during the course of the project; and
 - Annually a number of projects may be randomly selected for an external audit.
- The approval applies strictly to the protocol as stipulated in the application form. Would any changes to the protocol be deemed necessary during the course of the project, the project leader must apply for approval of these changes at the HSEC. Would there be deviated from the project protocol without the necessary approval of such changes, the ethics approval is immediately and automatically forfeited.
- The date of approval indicates the first date that the project may be started. Would the project have to continue after the expiry date, a new application must be made to the NWU-RERC via HSEC and new approval received before or on the expiry date.
- In the interest of ethical responsibility, the NWU-RERC and HSEC reserves the right to:
 - request access to any information or data at any time during the course or after completion of the project;
 - to ask further questions, seek additional information, require further modification or monitor the conduct of your research or the informed consent process;
 - withdraw or postpone approval if:
 - any unethical principles or practices of the project are revealed or suspected;
 - it becomes apparent that any relevant information was withheld from the HSEC or that information has been false or misrepresented;
 - the required annual report and reporting of adverse events was not done timely and accurately; and/ or
 - new institutional rules, national legislation or international conventions deem it necessary.
- HSEC can be contacted for further information via Lesetja.Motadi@nwu.ac.za or 018 289 2598.

The HSEC would like to remain at your service as scientist and researcher, and wishes you well with your project. Please do not hesitate to contact the NWU-RERC or HSEC for any further enquiries or requests for assistance.

Yours sincerely

Prof Lesetja Motadi

Chair NWU Health Science Research Ethics Committee (FAST-HSEC)



NORTH-WEST UNIVERSITY
YUNIBESITHI YA BOKONE-BOPHIRIMA
NOORDWES-UNIVERSITEIT

Faculty of Natural and Agricultural Sciences

Private Bag X2046, Mmabatho
South Africa 2735

Tel: 018 389-2051

Fax: 018 392-2052

Web: <http://www.nwu.ac.za>

Email: Helen.Drummond@nwu.ac.za

30 January 2018

TO WHOM IT MAY CONCERN

Dear Sir/Madam,

Data collection for research purpose

The Registrar,
Niger Delta University,
Amassoma, Bayelsa State
Nigeria

Dear Sir/Madam,

Data collection for research purpose

The bearer named E. A. Ekubo with student Number 29523389 is a PhD student in Computer Science. She is carrying out a research titled "Predictive system for characterizing low performance of Undergraduate students using machine learning techniques" in the department of Computer Science. Her research proposal has been approved by the university and ethics approval documents are been finalized.

She will require data for her research work. Please do not hesitate to give her the necessary assistance in her data collection efforts. She may conduct interviews and carry out observations. The data collected will be strictly used for the purpose of research.

Accept out kind assurances. Thank you for your assistance.

Yours sincerely

A handwritten signature in cursive script that reads "H. P. Drummond".

Prof Helen Drummond

Deputy Dean

Supervisor: Prof B.M. Esiefarienrhe



OFFICE OF THE VICE-CHANCELLOR
NIGER DELTA UNIVERSITY
Wilberforce Island, Bayelsa State
INTERNAL MEMORANDUM

From: The Vice Chancellor (Academics)

To: All Deans
All Heads of Department
All Faculty Officers
All Faculty Exam Officers

Date: February 6, 2018

DATA COLLECTION ASSISTANCE FOR RESEARCH

This is to inform you of the researcher Miss Ebiemi Allen Ekubo, a PhD student in Computer Science and Information Systems Department at North-West University, Mafikeng, South Africa. She requires assistance with collection of students' data for her research work. All information given must be anonymous and in line with university standards. Her research focus on developing models for improving students' academic performance in our university; thus, it would keep the university in good light.

Kindly assist her with all the required support.

Thank you.

Prof Donbebe Wankasi

Distribution

Faculty of Agriculture
Faculty of Arts
Faculty of Education
Faculty of Engineering
Faculty of Law
Faculty of Management Sciences
Faculty of Basic Medical Sciences
Faculty of Nursing
Faculty of Pharmacy
Faculty of Science
Faculty of Social Science



Office: 0183892451

FACULTY OF EDUCATION

Cell: 0729116600

Date: 9th March, 2020

TO WHOM IT MAY CONCERN

CERTIFICATE OF EDITING

I, Muchativugwa Liberty Hove, confirm and certify that I have read and edited the entire thesis, Predictive system for characterizing low performance of undergraduate students using machine-learning techniques by E A Ekubo, orcid.org/0000-0001-9348-5630, submitted for the degree of Doctor of Philosophy in Computer Science at the North-West University.

EA Ekubo was supervised by Professor B M Esiefarienrhe and co-supervised by Professor N Gasela of North-West University.

I hold a PhD in English Language and Literature in English and am qualified to edit such a thesis for cohesion and coherence. The views expressed herein, however, remain those of the researcher/s.

Yours sincerely

Professor M.L.Hove (PhD, MA, PGDE, PGCE, BA Honours – English)

