2017

A Dimensional student enrollment prediction model: case of Strathmore University

Bernard Ochieng Alaka
Fuculty of Information Technology (FIT)
Strathmore University

A Dimensional Student Enrollment Prediction Model: Case of Strathmore University

BENARD OCHIENG ALAKA

Masters of Science in Information Technology

2017

A Dimensional Student Enrollment Prediction Model: Case of Strathmore University

BENARD OCHIENG ALAKA

065739

Submitted in partial fulfilment of the requirements of the Degree of Master of Science in Information Technology at Strathmore University

Faculty of Information Technology

Strathmore University

Nairobi, Kenya

June, 2017

Declaration

I declare that this work has not been previously submitted and approved for the award of a degree by this or any other University. To the best of my knowledge and belief, the research proposal contains no material previously published or written by another person except where due reference is made in the research proposal itself.

Benard Ochieng Alaka - 065739

…….....

Date: …….....

Approval

The thesis of Benard Ochieng Alaka was reviewed and approved by the following:

Professor Ismail Ateya Lukandu

Associate Professor, Faculty of Information Technology

Strathmore University

Dr. Joseph Orero (PhD)

Dean, Faculty of Information Technology

Strathmore University

Professor Ruth Kiraka

Dean, School of Graduate Studies

Strathmore University

Abstract

The rate of student admissions within most Kenyan Universities has thus far been met with a corresponding uncertainty in budgetary allocation. Additionally, the increase of most applicants not being enrolled has led to lower institution yield. Due to the uncertainty of the quantity of students to be enrolled, planning and budgetary issues have arisen as stated earlier. Departments in charge of recruiting students are left to speculate the numbers likely to turn up. This in most cases is not accurate since it results into gaps in the allocated budgets and straining of resources. Currently, in Kenya, there is no institutions of higher learning that has a reliable means of predicting the expected institutional yield. Rather, academic management systems exist and are used to manage daily academic routines. These systems are served by transactional databases which are subject to being edited frequently and as such lack the capability of archiving histories of instances of the data within these databases; which makes them unsuitable for carrying out analysis on enrollment prediction. As such, a dimensional enrollment prediction model is proposed so as to aid in forecasting; not only of how many admitted students will be enrolled but also particular individuals who could show up for the purposes of follow-up activities. The inputs to the proposed enrollment prediction system were sourced from dimensional data stored in a data warehouse regarding to student details as per the admission as well as snapshot data of third party satisfaction index from accredited sources. The proposed system then transforms this data into dimensional data by adding a time variant to it and then passing the data through a neural network. The resultant model is then to be used in predicting students' enrollment. The proposed model was tested for accuracy using the precision, recall ratio and the F-score Measure. The model's accuracy was considerably high with an accuracy of 71.39% with a precision of 0.72. The average recall ratio was 0.71 and while F-score of 0.71 as well was obtained. In relation to some of the works reviewed the proposed model was a bit lower accuracy due the dataset used that was noisy as fetched from real student transactional databases.

Dedication

I foremost dedicate this research to God almighty, for his grace and mercies. I dedicate this document to my parents Samuel Alaka and Millicent Odhiambo; whose prayers and care always carried me through. I also dedicate it to my sibling Beryl Alaka, for keeping me on toes.

Acknowledgments

I would like to acknowledge my supervisor Professor Ismail Ateya Lukandu for his wise guidance. His wealth of experience and skill was so instrumental throughout this research. My appreciation also goes to Dr. Joseph Orero and Dr. Vincent Owenga for their very useful insights.

I would also like to appreciate my colleagues, Sharon Mugambi, Eunice Manyasi, Patrick Kiplimo and Titus Tunduny who helped review my thesis. Much appreciation also go to my friend Wallace Muchiri for his encouragement and generosity.

Finally, I highly appreciate the support of my bosses Mr. Patrick Shabaya and Mr. Johnson Muthii who encouraged me to pursue my postgraduate degree relentlessly. God bless them for their generosity and mentorship.

Table of Contents

List of Figures

List of Tables

List of Equations

Abbreviations/Acronyms

ANN	-	Artificial Neural Networks
CRISP-DM	-	Cross Industry Standard Process for Data Mining
ETL	-	Extraction, Transformation and Loading
HESEPS	-	Higher Education Student Enrollment Prediction System
ID3	-	Iterative Dichotomiser 3
MAPE	-	Mean Absolute Percentage Error
MLP	-	Multilayer Perceptron
MSE	-	Mean Squared Error
RMSE	-	Relative Mean Squared Error
RAE	-	Relative Absolute Error
SDLC	-	System Development Life Cycle
SSI	-	Student Satisfactory Inventory
SVM	-	Support Vector Machines

Definition of Terms

Dimensional Data - Data stored within the facts and dimensions of a data warehouse (Kimball Group, 2005)

Neural Networks - Biologically inspired analytical techniques, capable of modeling extremely complex non-linear functions and used machine learning tasks (Delen, 2010).

Institution Yield - The percent of students who choose to enroll in a particular college or university after having been offered admission (Arcilla, 2012).

Date Warehouse – A subject-oriented repository that takes snapshot of transactional databases and stores in forms of facts and dimensions for analytical purposes (Ralph Kimball, 2002).

Time variant – An attribute within the data warehouse that shows the date on which a snapshot of the transactional database was taken (Furlow, 2001).

Chapter 1 : Introduction

1.1 Background

In the past years, the number of admitted students in Kenyan Universities has always superseded the number of enrolled students. Following the recruitment process, many students are normally served with admission letters, with the different schools of faculties anticipating an almost equal number to show up. However, not every student who gets admitted ends up being enrolled for various reasons as is discussed in section 2.4. The minimum requirement for admissions in most Kenyan Universities entails at least passing the requisite entry exams as well meeting the minimum entry requirement such as high school mean grade. Enrollment on the other hand, as is the standard in most institutions, occurs after the student pays an amount of fee and registers for a number of given courses or units offered by the institution.

The ratio of numbers of students enrolled to the number of students admitted is known as institutional yield (Arcilla, 2012). Institutional yield is considered to be among the most reliable indicators of institutional quality and competitive standing. This is primarily because of the consistent research regarding the benefits of a college degree to individuals and to society (Victoria University, 2013). Institutional yield as mentioned by University of Pennsylvania (2013), may appear to be a simple rate and a commonsense indicator. However, the reality is that calculating and interpreting it is far more complex and analytically challenging than one might think; and in the end may lead to misrepresenting the reality about how an institution is doing.

The use of management systems within the institution makes it quite difficult to analyze data that relates to student enrollment in relation to the numbers of students that are admitted. Data in the transactional databases that server these management systems are highly normalized and very atomic, and thus can only exist in one consistent state. This makes it hard to show the same unit of data across different time dimensions. It thus gets challenging to pull logically demarcated analysis from the data. Additionally, some data that regards to student reasons for not enrolling are until recently only captured through questionnaires or Google forms and analyzed separately from the other institutional yield analyses. Coming with up with satisfaction indexes, so as to get an opinion of the students about the university, has been a major problem since they are conducted by external bodies and not the universities themselves.

So far, only a few institutions worldwide have adopted the enrollment prediction as a crucial process for aiding their decisions. The approach used by these institutions have their algorithms informed by minimal parameters that are only provided for by the prospective students during the admissions process such as age, high school score, parents occupation, fee guarantor among many other parameters. This makes the algorithm biased to the student data only as a determinant factor for enrollment. As outlined by Zaytsev (2011), other metrics such as national ranking of a university have become increasingly important in the past few years as schools and students have started paying more and more attention to rankings that are released yearly. He further states that Empirical evidence suggests that these rankings influence student behavior in the college admissions process. As such an enrollment prediction model that uses dimensional data taken from both student details and satisfactory index is proposed to fill up for these shortcomings.

The dimensional data approach of organizing student data, unlike the legacy format of storing information in institutional transactional databases, provides for the ability to store multiple instances of data. An instance of a student's record existing within the transactional database may undergo a series of updates. This in the long run makes it difficult to track the changes made to this record and perhaps draw out analytical conclusions. A student for instances who wishes to take a course in Commerce may at some change their choice to a course in Hospitality. Upon update, the original form of the data is lost. One of the simplest advantages of employing dimensional data to this scenario is the ability to see snapshots of the given student's data before and after the update was applied. This coupled with the time variant capability of the dimensional data, is sufficient enough to be used in the proposed enrollment prediction model. This is because, the time variant is connected to a series of external events that could be influencing enrollment that may not be well captured in the snapshots taken of the transactional databases.

The prediction of student enrollment is seen as an attempt towards narrowing the budgetary gaps that arise out of uncertainty caused by the unknown number of admitted students likely to show up in class as enrolled students. It is evident with the assurance of the numbers likely to turn up, to a satisfactory degree, that the relevant management and authorities within the

University will be empowered to make informed budgetary decisions with regards to how much resources is to be allocated. Additionally, the proposed model shall offer drill down capability of finding out the individual student most likely not to show up for purposes of follow up by the admissions or recruitment departments.

1.2 **Problem Statement**

Most administrators within the universities have over years relied on a speculation based approach in the process of forecasting the number of students to be enrolled. This approach works conducts a trend analysis on enrolment coupled with admission experience, and then tries to gauge what the next enrollment would be in terms of numbers. This has so far been erroneous and has led to a rippled negative effects such as increase the number of intakes over a relatively shorter span of time to fill up for the budgetary gaps (Owuor, 2012). This has then resulted into the straining of the available fixed resources within the campus, thus negatively affecting the quality of the learning experience.

This in the long run has compromised on the quality of the institution in terms of the quality of students being enrolled (Henriques, 2015). The currently existing systems have only served well in the management of data and the production of periodical reports based on the data that exists within the transactional databases that are prone to being edited frequently. This frequency of editing information interfered with the historic aspect of data that is necessary for coming up with a data set rich enough for carrying out analysis on the number of students that are most likely to get enrolled.

Therefore a prediction system is proposed to help predict the admitted individuals who are likely not to get enrolled and follow up with appropriate action so as to improve on enrollment rate. The proposed system was not only able to capture the students' dimensional data but also incorporate third party satisfaction index and co-relate the impact of this index to the enrollment statistics for a more informed parameter base for the related machine learning algorithm.

1.3 **Research Objectives**

i. To investigate factors influencing student enrollment in Universities
ii. To review existing enrollment prediction models for Universities
iii. To propose an enrollment prediction model for predicting student enrollment for Universities in Kenya

iv. To test the proposed enrollment prediction model in Strathmore University

1.4 **Research Questions**

i. What factors influence student enrollment in Universities?

ii. What are the prediction models used in determining student enrollment in Universities?

iii. How can the proposed model be used to improve the enrollment prediction in Kenyan Universities?

iv. How efficient is the proposed system?

1.4 Justification

The use of data to produce ad-hoc reports is but a budding process in the Kenyan educational sector. Some of the institutions that have adopted business intelligence in their daily operations, are on the other hand yet to find better ways of capturing the massive data within their databases, and produce exhaustive reports as relates to the institution yield, which are the major measures used globally for ascertaining the competitive edge of institutions of higher learning (Ministry of Education, Science and Technology - Kenya, 2015). This work thus aims to come up with a solution of predicting institutional yield, based on the data availed; to assist in making crucial institutional decisions that would enable these institutions gain competitive edge and also improve soundness in decision making.

1.5 Scope and Limitation

This research is limited to the prediction of students to be enrolled. Of interest to this study is not to find out why these students are likely to be enrolled but to classify those likely to show up in class against those likely not to show up. The research is also limited to short term prediction as opposed to long term forecasting of enrollment trends.

Chapter 2 : Literature Review

2.1 Introduction

The enrollment trends and the importance of these trends were reviewed so as to gain more insight on the research problem. Challenges faced by institutions' lack of an enrollment prediction system were also studied. Works relating to predictions such as models used in the prediction of universities retention rate, attrition rate, and graduation rate were reviewed and the architecture critiqued as well. As a result, an improvement of these models was presented as a solution for enrollment prediction.

2.2 Enrollment Trends in Kenyan Universities

As reported by Kenya National Bureau of Statistics (2015), there are about thirty universities in the country today that have between 40,000 and 60,000 students qualifying for admission to public universities, which only absorb 10%. Only 3% being absorbed by Private Universities as shown in Table 2.1 and Table 2.2 which depict the total enrolled students for each academic year. This study goes ahead to point out that the growing numbers, in both public and private universities, that have been established in the country for the past decade would be deemed as good news; since with increase in the number of universities the wastage of students who attain university entry grades would be reduced.

Table 2.1: Public Universities Aggregate Enrollment Trend (Kenya National Bureaus of Statistics, 2015)

Fields	2009/10	2010/11	2011/12	2012/13	2013/14+	2014/15*
Undergraduates..	108,528	134,395	141,764	170,417	264,649	323,434
Postgraduates..	7,054	8,735	16,153	24,417	38,318	44,274
Other, i.e. Diploma, etc.	7,118	7,796	5,904	6,856	26,792	32,510
Total..	**122,700**	**150,926**	**163,821**	**201,690**	**329,759**	**400,218**

Table 2.2: Private Universities Aggregate Enrollment Trend (Kenya National Bureaus of Statistics, 2015)

Private Accredited	2009/10	2010/11	2011/12	2012/13	2013/14	2014/15*
Daystar University+	3,793	5,915	4,049	5,431	4,061	4,170
Baraton University	2,019	3,149	2,155	2,344	2,092	2,198
Catholic University	2,019	3,149	2,155	3,647	5,211	7,025
U.S.I.U.	4,590	7,158	4,899	5,206	5,525	5,534
Scott Theological College	131	204	140	255	365	588
Agha Khan University	179	279	191	503	718	500
Strathmore University+	2,341	3,651	2,499	5,811	5,821	6,848
Kabarak University+	1,126	1,756	1,202	1,215	3,027	3,055
Nazarene University	1,285	2,004	1,372	1,932	2,760	3,935
Methodist University+	2,426	3,783	2,589	11,203	11,849	11,859
Kiriri Women University of Sc.Tech	180	281	192	124	177	253
TOTAL	**20,089**	**31,327**	**21,443**	**37,672**	**41,606**	**45,965**

However, this has not been the case since it is reported that most universities recorded a decline in student enrollment (Kenya National Bureau of Statistics, 2015). Some attracted only about 50% or as low as 11.62% of previous year's enrollment. The trend is worse in private Universities where admission rates are high while the enrollment is very low. This trend as further explained by Gudo (2014) has continued and is believed to have been accompanied by a decline in quality of university education. Quality in this sense is compromised when crucial decisions such as resource allocation decisions are made based on misguided facts concerning enrollment of students. The contrast in enrollment between public and private universities in Kenya as exemplified above is explained by Ibrahim (2012) as being attributed to a variety of reasons. He states as his reasons the late introduction private university education in Kenya, low enrollment capacity of private universities, narrower range of courses offered in private universities, high tuition and accommodation and transportation fees in private universities.

2.3 Significance of Enrollment Management in University Resource Planning

Enrollment management as presented by Schuttinga (2011) is the use of enrollment data to track the rate of enrollment within the institutions of learning. Enrollment as he further explains has recently been applied less in planning, managing, and teaching. In his paper, he seeks to emphasize that besides serving higher education's consumers, accountability should also serve those who plan and manage enrollments. Enrollment management according to Schuttinga (2011) is improved by use of a forecasting and simulation model in which "performance" measures such as enrollment forecasting, enrollment management, and retention - play a major role.

According to Bontrager (2004), knowing enrollment trends is the central component of effective budget and program planning. This is so because Student enrollment directly translates into fiscal income, which is not only fundamentally important to budget but also to program, and personnel planning. Accurate enrollment forecasts are crucial for colleges and universities to remain competitive while inaccurate enrollment forecasts can lead to poor allocation of funds and resources (Chau-Kuang Chen, 2008).

Enrollment management has also proven to be an effective avenue for ensuring the establishment of clear goals for the number and types of students needed to fulfill the institutional mission, increasing process, and organizational efficiency, creating a data-rich environment to inform decisions and ensuring sound evaluation of institutional competitive strategies (Bontrager, 2004). This is recently the case as universities have found the need to appeal to an ever-increasing and diverse student base through successful branding and marketing. This has resulted in a more in-depth market research by recruitment bodies within the universities which majorly sources its data sources from the enrollment data (Hanover Research, 2014).

It is also important to capture enrollment data not just for budgeting and planning purposes or for marketing but also to add value to the individuals who are admitted but fail to gain enrollment for a variety of reasons. An enrollment prediction system, for instance, would come in handy in this instance as it would not only forecast the numbers that are most likely to be enrolled but actually highlight the individuals who are most likely not show up. An extra

effort such as follow-up through phone calls, or granting of financial assistance can then be dispensed there forth by the appropriate authorities within the institutions.

2.4 Factors that influence enrollment in Universities

In a report done by Human Development Resource Center (2010), the factors that affected the enrollment rates in Kenyan universities were sourced from various other reports. The reasons for not enrolling generally gravitated around the lack of funds by the admitted students, distance from the learning institutions, domestic commitments and ignorance of parents or guardian. To arrive at this reasons, the Human Development Resource Center (2010), records that most of the information was found on a child-to-child survey run in Kenya.

In another research conducted by National Research Report (2012), enrollment within American universities is reported to have been affected by a variety of reasons as well. First are cost issues, which addresses how important tuition fee is, in the student's decision to enroll. Students also take into consideration financial aid options offered by the institution. Others factors included academic reputation, the size of the institution, a recommendation from family or friends, geographic setting, campus appearance and even personalized attention offered by the institution during the admission process. This he summarizes in the bar graph in Figure 2.1

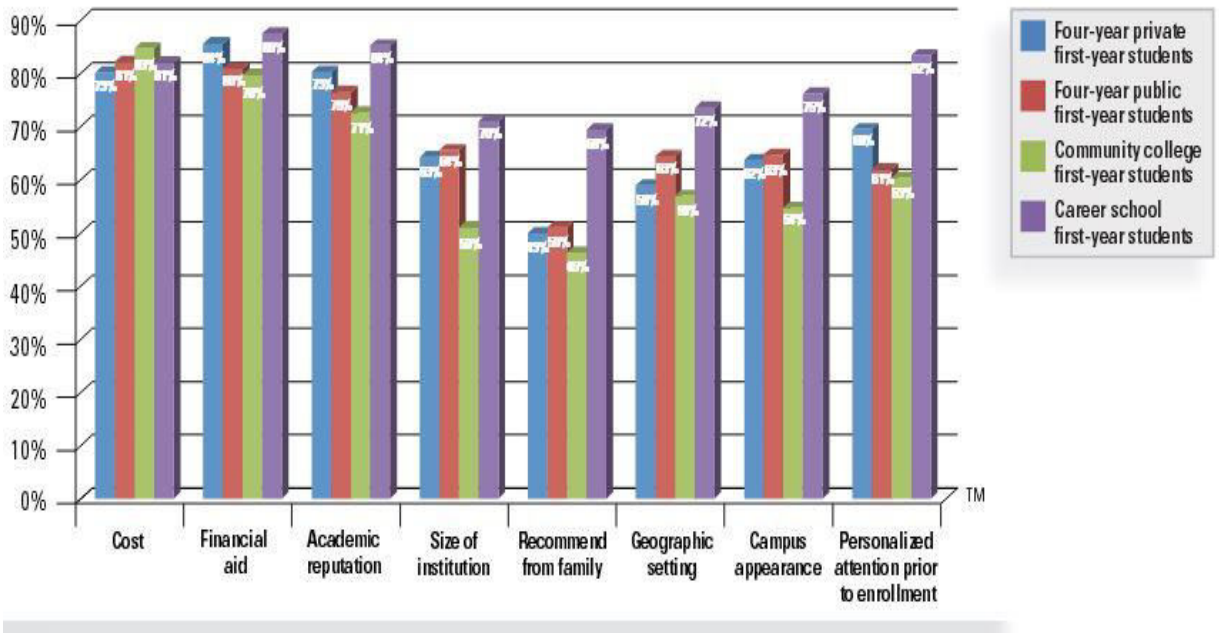


Figure 2.1 Factors that contribute to low enrollment (National Research Report, 2012)

Figure 2.1 shows the results that were arrived following annual test that is conducted every year by hundreds of campuses known as the Noel-Levitz Student Satisfaction Inventor, which is administered to the students. In addition to more than 70 items rated for importance and satisfaction on the general student experience, the SSI includes nine items that address factors in a student's decision to enroll (National Research Report, 2012).

In order to arrive at the reason for either enrolling or not enrolling by, surveys of different kinds has most often than not been used. For instance, for the past 10 years, many institutions have adopted the use of Noel-Levitz Student Satisfaction Inventory (SSI) as part of a strategic enrollment management initiative. This instrument works by using a gap analysis technique to array students' satisfaction against their perceived importance of various aspects of the college experience. A newer variant of Noel-Levitz is called the Adult Student Priorities Survey which is tailored to the needs of adult learners.

However as established by Pbnny et al (1990), the use of surveys though efficient in some scenarios has proven to be very inconclusive due to the total survey error perspective which recognizes that the ultimate goal of survey research is to accurately measure particular constructs within a sample of people who represent the population of interest.

It is further ascertained that the total survey error perspective, disaggregates overall error into four components: coverage error, sampling error, nonresponse error; and measurement error. Coverage error is the bias that can result when the pool of potential survey participants from which a sample is selected does not include some portions of the population of interest. Sampling error is the random differences that invariably exist between any sample and the population from which it was selected. Non-response error is the bias that can result when data are not collected from all of the members of a sample. Measurement error as all distortions in the assessment of the construct of interest, including systematic biases and random variance (Pbnny & Vissbr, 1990).

The use of a more robust instrument that would not only rationalize the enrollment trends but also predict the same with a minimal error of biasness or exclusion of data sets would be very highly recommended. This can only be met through data mining which entails applying data analysis and discovery algorithms that, under acceptable computational efficiency limitations, produce a particular enumeration of patterns (or models) over the data (Ozer, 2008).

2.5 Enrollment Prediction Models and Algorithms

For the purposes of this research the approaches reviewed enrollment prediction were classified into two: empirical models and machine learning algorithms.

2.5.1 Empirical Models

Empirical models have largely been used in institutions of higher learning to forecast the number of students likely to be enrolled. Such methodological approaches include logistic regression and discriminant analysis which are majorly used in retention studies and also in identifying factors that contribute to student dropout and enrollment as well. Generally, most empirical models work by following the course of problem definition and data collection, model formulation, model verification and model implementation (Arcilla, 2012).

After the selection of the model to be used for prediction, the next step entails selecting variables to be used, selecting the form of the equation relationship and also estimating the values of the parameters in the chosen equation. After this, the next advisable step is verification and validation which involves comparing forecasts with historical data for the intended process to be forecasted. Different error measures such as MAPE, RAE, and MSE are normally used for

validation. The kind of error measure used to show the accuracy of a model normally impacts a lot on the conclusion of the forecast model to be chosen (Kalekar, 2004).

As further stated by Kalekar (2004), a category of forecasting models known as time series forecasting comes with specific prediction models that suit enrollment prediction. He goes ahead to assert that time-series forecasting assumes that a time series is a combination of a pattern and some random error. The goal, therefore, entails having to separate the pattern from the error by understanding the pattern's trend, its long-term increase or decrease, and its seasonality; which is the change caused by seasonal factors such as observable fluctuations in use and demand. Several methods of time series forecasting are available such as the Moving Averages method, Linear Regression with Time and Exponential smoothing being the most commonly used in prediction tasks (O. Anava, 2013).

Since the model proposed in this research bases its dataset from historical data, empirical models that are futuristic were not studies. Only simple moving average as an empirical model was reviewed since its dataset is informed by historical data.

2.5.1.1 Use of Simple Moving Average for Enrollment Prediction

Simple moving average, in particular, was used by Arcilla (2012) for enrollment prediction in Mindanao State University – Ilagan Institute of Technology. This model worked by predicting the future number of enrollment through calculating an average of enrollment numbers from a specified number of prior enrollment. Each new forecast drops the demand in the oldest period and replaces it with the demand in the most recent period. The formula for simple moving average is shown in Equation 2.1

$$T_t = 1/3(Y_1+Y_2+Y_3) \quad \text{Equation 2.1: Simple Moving Average function}$$

Where:

Y_1 – is the 3rd value from the value to be forecasted

Y_2 – is the 2nd value from the value to be forecasted

Y_3 – is the 1st value from the value to be forecasted

As indicated by Gor (2002), the simple moving average is normally best applicable in instances where a rate under study is neither growing nor declining rapidly. It becomes useful in

such instances since it is able to remove random fluctuations for forecasting. Moving averages as further explained are frequently centered as they use the central measure of tendency. Despite this, it is still possible to use this model to predict the future using past data. An illustration for the forecasting of the enrollment numbers of say the month of June is arrived at a five-month moving average, which takes the average of January, February, March, April and May. While to forecast for July, the averages of February, March, April, May and June would be considered.

Although it is important to select the best period for the moving average, there are several conflicting effects of different period lengths. The longer the moving average period, the more the random elements are smoothed. But if there is a trend in the data either increasing or decreasing the moving average has the adverse characteristic of lagging the trend. Therefore, while a shorter time span produces more oscillation, there is a closer following of the trend. Conversely, a longer time span gives a smoother response but lags the trend hence less accurate (Gor, 2002).

2.5.2 Machine Learning Algorithms used in Enrollment Prediction

With the onset of machine learning, the empirical models are slowly being replaced by machine learning algorithms that are being used for prediction tasks. These algorithms are more robust and less susceptible to noise within the provided datasets. The most common machine learning algorithm that has been used for prediction of university enrollment and student retention rates are decision trees and support vector machines.

2.5.2.1 Use of Decision Trees for Enrollment Prediction

Decision trees are increasingly becoming a preferred algorithm used when it comes to prediction tasks due to their intuitive characteristics. Popular decision tree algorithms include ID3, C4.5, and C5 decision trees. So far, very few Universities use decision trees for purposes of enrollment prediction. Rather, most have resorted to using this algorithm in retention rate prediction and graduation grade prediction as explained by Kalpesh et al (2013).

Decision trees are classifiers that are expressed as a recursive partition of the input space based on the values of the attributes. The decision tree consists of an internal node that splits the

instance space into two or more subspaces according to a function of the input attribute values. Each leaf is then assigned to one class that represents the most appropriate or frequent target value. Instances are classified by traversing the tree from the root node down to a leaf according to the outcome of the test nodes along this path. Each path can be transformed then into a rule by joining the tests along this path (Abdul et al, 2012).

So as to select the most suitable algorithm to split the decision tree, the ID3 algorithm uses some information as a splitting criterion. This criterion uses an entropy index that measures the degree of impurity of the certain labeled dataset. For a given labeled dataset S with some examples that have n (target values) classes {c1, c2… cn), entropy index (E) is defined in Equation 2.2

$$E(S) = \sum_{i=1}^{n} p_i * \log(p_i), \quad p_i = \frac{|S_{c_i}|}{|S|}$$

Equation 2.2: ID3 Entropy function

Where S_{ci} is the subset of the examples that have a target value that equals to ci and Pi is the probability of occurrence of an instance within a dataset. Entropy (E(s)) has its maximum value if all the classes have equal probability. Therefore to select the best attribute for the splitting of a certain node, information gain measure is used. To get the information gain, Gain (S, A) of an attribute say A, the function in Equation 2.3 is used.

$$Gain(S, A) = E(S) - \sum_{v \in V(A)} \frac{|S_{A=v}|}{|S|} E(S_{A=v})$$

Equation 2.3: ID3 Information Gain function

Where E(S) is the entropy index for dataset S, V (A) is the set of all values for attribute A. This algorithm is however prone to many flaws as well. However those pertinent to enrollment prediction are that the algorithm is more sensitive to noise. This implies when increasing, decreasing or modifying the training set, the decision tree will change as well. Additionally, decision tree algorithms especially ID3 algorithm do not consider attribute correlation (Xie et al, 2010).

Kalpesh et al (2013) tested these algorithms in classifying admitted students into two categories, whether or not they would get enrolled. The dataset was was limited to data gathered

from transactional databases. The accuracy attained in their validation was high enough due to the nature of data that they used that was already clean. Decision trees however have to built from time to time should the classification rules change. This makes it quite challenging to implement and maintain.

2.5.2.3 Use of Support Vector Machines for Enrollment Prediction

Support vector machines as defined by Delen (2010) belong to a family of generalized linear models which achieves a classification or regression decision, based on the value of the linear combination of features. The mapping function in SVMs can be either a classification or a regression function. For classification, nonlinear kernel functions are often used to transform the input data to the original input space.

The maximum-margin hyper planes are then constructed to optimally separate the classes in the training data. Two parallel hyper planes are constructed on each side of the hyper plane that separates the data by maximizing the distance between the two parallel hyper planes.

Table 2.3: Accuracy of Machine learning algorithms for enrollment prediction (Delen, 2010)

		ANN(MLP)		DT(C5)		SVM		LR	
		No	Yes	No	Yes	No	Yes	No	Yes
Confusion Matrix	No	2309	464	2311	417	2313	386	2125	626
	Yes	781	2626	779	2673	777	2704	965	2464
SUM		3090	3090	3090	3090	3090	3090	3090	3090
Per-class accuracy		74.72%	84.98%	86.50%	86.50	74.85%	87.51%	68.77%	79.74%
Overall accuracy		79.85%		80.65%		81.18%		74.26%	

The main assumption made here was that the larger the margin or distance between these parallel hyper planes, the better the generalization error of the classifier will be. Other algorithms such artificial neural network, decision trees and logistic regression performed less accurately as compared to support vector machines. Table 2.3 shows the performance of the above machine learning algorithms in a study conducted by Delen (2010) who attempts to perform enrollment prediction using Support Vector Machines. Support vector machines are however as compared to neural networks very sensitive to noise and thus could not perform well for this research where real student data, which was noisy was used.

2.5.2.4 Institutional Data Warehousing

A data warehouse is a large repository of well-organized data gathered from a variety of sources such as spreadsheets and databases that is non-volatile in nature and is used to aid decision-making processes within an organization (Kimball Group, 2005). The implementation of the data warehouse within in institutions of higher learning has been successful of late and the quality of data within the data warehouses are of quality due to the fact that they have undergone cleansing (Furlow, 2001).

Data warehouses, unlike transactional databases, organizes data into dimensions and facts. Dimensions as outlined by Kimball Group (2005), explains the "*who, what, where, when, why*, and *how*" context surrounding a business process. In most cases, a dimension should be single-valued when linked, joined or associated with a relevant fact table. This thus makes dimension tables the core of a data warehouse since they contain descriptive attributes necessary for business analytics.

Fact tables, on the other hand, are the measurements that result from a business process event and are almost always numeric (Ralph Kimball, 2002). Fact tables are used to store data that corresponds to the physical observable event(s) within a business process. The join of a dimension table with a fact table result into a multidimensional data mart. This multidimensionality is attributed to the time variant added to the data set. The time variant entails data whose context is relevant to some moment in time and is existent in three forms: continuous time span, event-discrete, and periodic discrete data (Inmon, 2005). For this study, the event-discrete time variant was used to add dimensionality to the admitted students' data. This is because, many events during the academic calendar year have a role to play in the enrollment patterns of the admitted students (Arcilla et al., 2012). To achieve this, a calendar date dimension was attached to the admissions fact table to allow navigation through familiar dates, months, fiscal periods, and special days on the calendar.

The calendar date dimension makes it easy to point to a date, from a fact table that is already described within the calendar date dimension. This would typically include many attributes such as week number, month name, fiscal period, national holiday indicator as well as a description detailing what exactly happened on that given date, say advertisement of a given

course (Inmon, 2005). Upon adding the dimensionality the data, the analyzed data can then visualized on dashboards in form of graphs, cross-tabs, and charts for purposes of aiding decision making.

Having an institutional data warehouse is thus thought to be beneficial in the sense that it allows for historical intelligence since data stored in transactional databases, data stored in a data warehouse is bound to a particular time frame. This is referred to as freezing of data. This feature allows for the capability of accessing snapshots of data in relation to a specified time frame. This thus makes it easier for the analyst to track and map the historical data of different departments in the University for Strategic planning (Kimball, 2002).

Another major merit of institutional data warehouses is that it allows for standardization of data. Ordinarily with different systems operating within the university, the database structures and ultimately the format of data housed therein also differs. . Following careful cleaning of the data through the ETL process, a data warehouse thus seeks to dispel the aforementioned by improving both the quality and consistency of data loaded into it. This impacts on the machine learning algorithms used since it helps in reducing noise in the data (Xu, 2007). Other prospective benefits of having data in a standardized format would include the possibility of highlighting new links in relationships between entities of interest and maintenance of data integrity. The diagram in Figure 2.2 shows the general architecture of an institutional data warehouse.

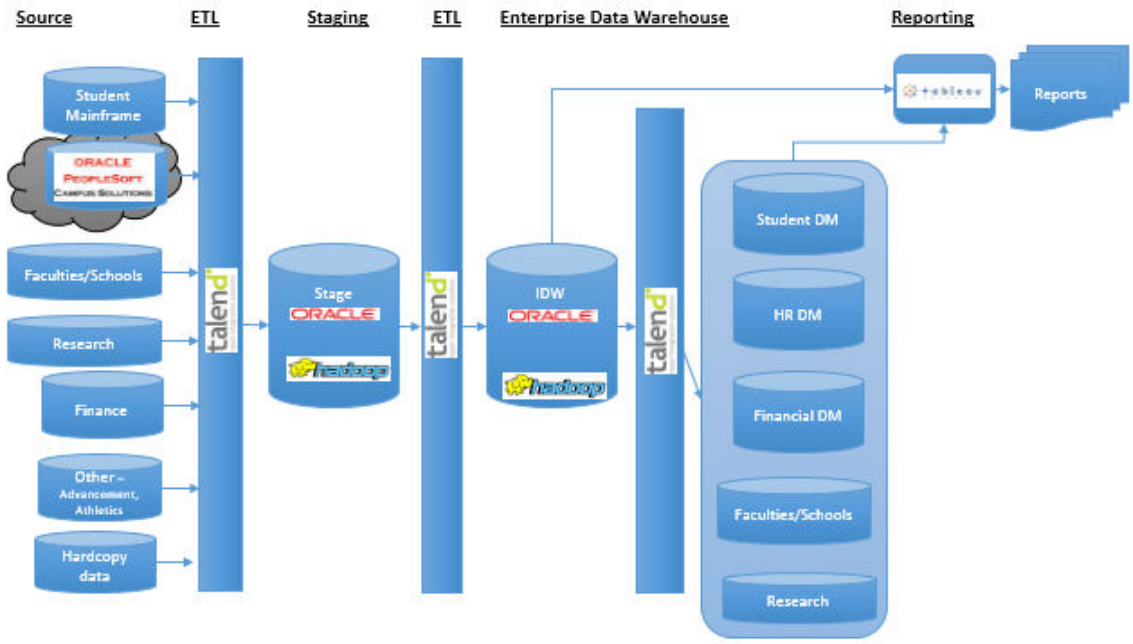


Figure 2.2: Data Warehouse Architecture (Xu, 2007)

2.5.2.5 Enrollment Prediction Models

Borena et al (2014) proposed an enrollment prediction system that is based on a predictive model, which determined the number of higher education students' enrollment at department level ahead of time using data mining approaches. This model worked by comparing three different algorithms which are: Decision tree (J48 Classifier), Bayesian Classifier (Naïve Bayes) and Neural Network (Multilayer Perceptron). The algorithms were evaluated and compared using model comparison technique such as confusion matrix, classification rate and area under the ROC curve in each of the experiment.

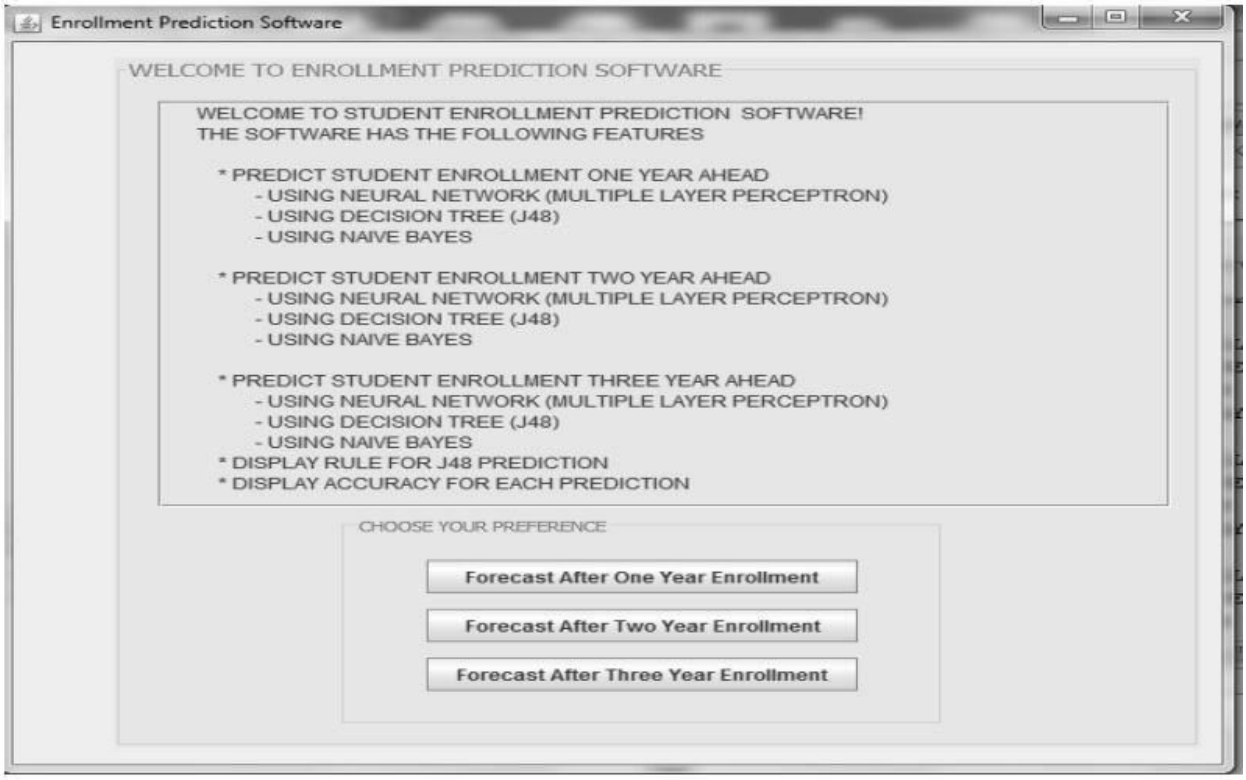


Figure 2.3: Higher Education Student Enrollment Prediction System (Borena et al, 2014)

The framework used here is known as CRISP-DM and was used to conduct systematic data mining analysis. Its stages are organized, structured and defined into six steps namely; business understanding, data understanding, data preparation, modeling, evaluation and prototype development. This system starts by preprocessing data to manage outliers, missing values and reduce the level of dispersion between the variable in the data set. Data sets at this stage are divided into two parts as the "training set" and "test set" using 10-fold Cross Validation. After this stage, predictive data mining model is used for each data using three data mining techniques: decision tree, neural network, and bayesian classifier. After data mining is complete, four model adequacy criteria are used to measure the performance and adequacy of the prediction model.

The system predicted students' enrollment not only by using the best algorithm but using all of the three data mining algorithms, which means the experts had different alternatives. This was made possible by the java user interface in Figure 2.3 that allowed a user to choose their

algorithm of preference. The adequacy criteria carried out to establish the accuracy levels of the used algorithms revealed that the use of Neural Networks yielded more accurate prediction of enrollment followed by the J48 decision tree classifier algorithm as presented in Table 2.4

Table 2.4: Accuracy for algorithms used in HESEPS (Borena et al, 2014)

Model	Accuracy	True Positive Rate	F-Measure	ROC Area	Time
J48	85.42%	0.854	0.853	0.934	0.03
Naïve Bayes	72.22%	0.722	0.715	0.874	0.02
Multilayer Perceptron	91.9%	0.919	0.919	0.969	32.72

An almost similar approach was followed by Padmapriya (2012) who attempted to predict enrollment using classification algorithms. This approach was not as structured as the one used by Borena et al (2014), rather, it sought to study the best classification algorithm that would be used for forecasting enrollment in a College in India. The samples for the classification for this system was drawn from Government Arts College for Women, Pudukkottai and was randomly partitioned into two independent sets, a training set, and a test set. The training set was used to derive the classifier, while the accuracy was estimated using the test set. The training samples had its attributes carefully chosen and analyzed before data mining commences. The classification rules learned from the analysis of data from existing postgraduates were then used to predict whether a student will join a given course or not.

Two classification algorithms were used for this system namely decision tree algorithm and Naive Bayesian Classifier algorithm. These algorithms as outlined by Padmapriya (2012) were chosen because they are considered as “white box” classification model and can be used directly for decision making. The highest accuracy in this study was achieved by the decision tree model with a classification accuracy of 93.33%.

2.6 Conceptual Framework

Data was first extracted from the existing transactional databases. The data then underwent transformation such as alteration of data types, encoding and denormalization of the relations extracted. The transformation stage also entailed including an aspect of dimensionality

to the data by adding a time variant to the snapshots taken. The transformed data was then loaded into an already designed data warehouse that was made up one fact table and one dimension. The data in the data warehouse was then fetched by a parameterization engine that was meant to select attributes that actually have an impact on the class attribute; this is known as feature selection. The undesired attributes were filtered out with the relevant ones proceeding to the machine learning algorithm.

The machine learning algorithm used was the artificial neural network. A feed-forward multilayer perceptron approach was used for learning. The data upon being fetched from the warehouse was first divided into a training set and testing set using a 10-fold cross validation rule. The data was then normalized and scaled into values ranging between -1 and 1. Appropriate features were then used to classify the attributes in relation to the provided enrolled classes decoded as either -1 or 1. The results were then validate using the validation set to ascertain the accuracy of the learning algorithm. A web application user interface was provided to enable the intended users to input new data instances and get the proper classification either in bulk or per individual student. The conceptual flow is further detailed in Figure 2.4

.

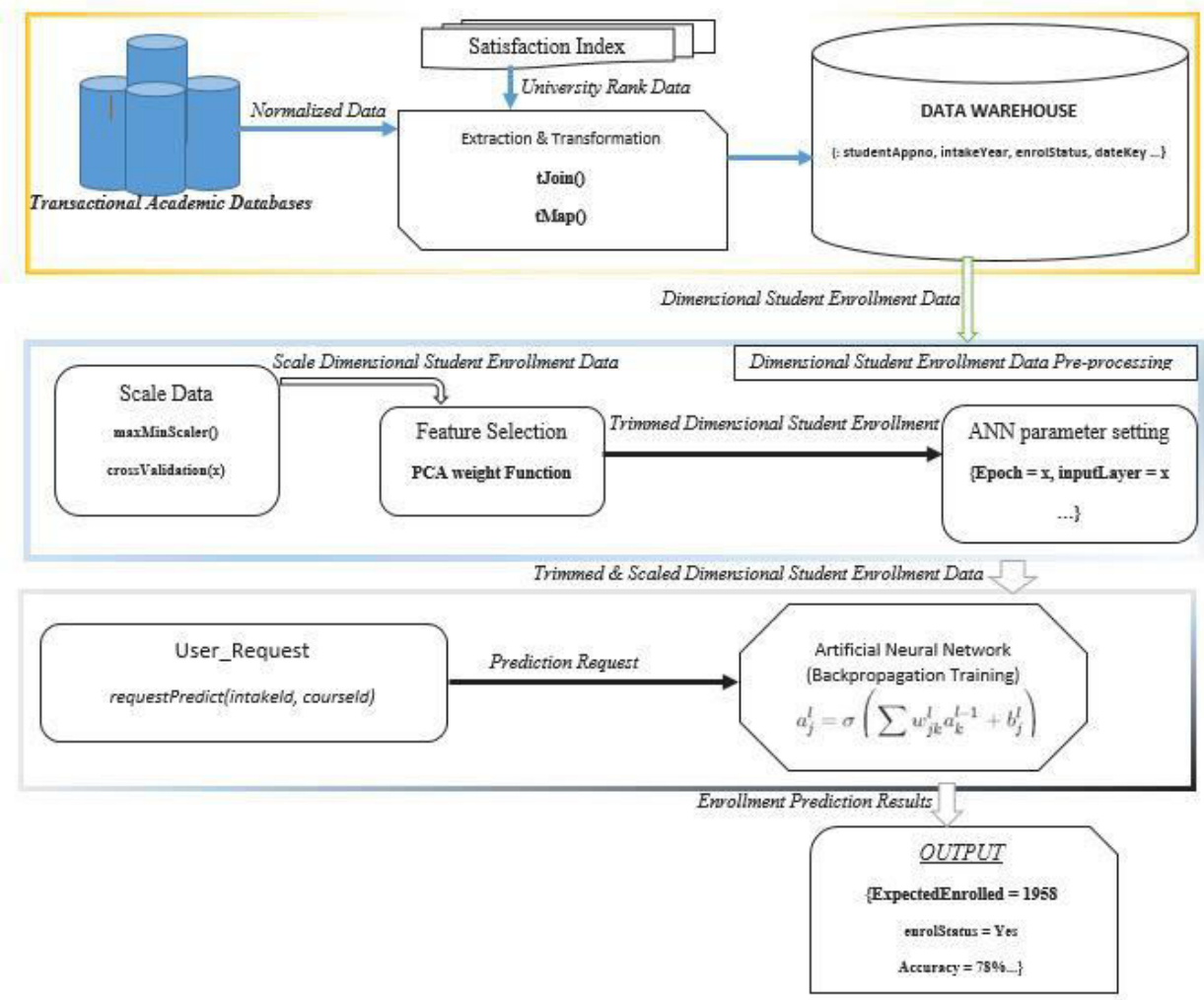


Figure 2.4: System Conceptual Framework

Chapter 3 : Research Methodology

3.1 Introduction

Research methodology is of importance as it facilitates the smooth sailing of the various research operations, thereby making research as efficient as possible, yielding maximal information with minimal expenditure of effort, time and money (Kothari, 2004). It is important to identify the most suitable methodology that suits a given research. This is very crucial in an information system, whose designs may vary from one system to another. And as such the selection of an improperly fitting methodology would not only have a lagging time effect on the process of coming up with the system but also impair the potential ability of the system solving the problem that it was intended for.

For this research, it was therefore appreciated that the use of both primary and secondary sources of information helped in identifying the approaches used in enrollment prediction and the accuracy of such approaches. The system design approach used was the prototyping design. This design was selected because of the final output of this research which was a prototype rather than a complete system. Also being a machine learning based system, algorithms shall from time to time be tuned so as to learn data in the most efficient ways. The data sources as well shall from time to time be validated to ensure that correct data passes through the algorithms.

3.2 System Development Methodology

Most information systems are featured to have very dynamic expectation from the users or individuals using them. As such, the development is presumably a continuous process so as to keep these systems relevant as possible to the ever-changing environment. The proposed system that dealt with parameterization of enrollment factors, also faced the same scenario, where new factors relating to the student enrollment shall always arose as advised by the experts in the field of students' admissions. Thus to come up with the most appropriate system development methodology, a criteria outlined by Alan et al. (2012) in Table 3.1 was used.

Table 3.1: Criteria for Selecting a Methodology (Alan, Barbara & Roberta, 2012)

Usefulness in Developing Systems	Waterfall	Parallel	V-Model	Iterative	System Prototyping	Throwaway Prototyping	Agile Development
with unclear user requirements	Poor	Poor	Poor	Good	Excellent	Excellent	Excellent
with unfamiliar technology	Poor	Poor	Poor	Good	Poor	Excellent	Poor
that are complex	Good	Good	Good	Good	Poor	Excellent	Poor
that are reliable	Good	Good	Excellent	Good	Poor	Excellent	Good
with short time schedule	Poor	Good	Poor	Excellent	Excellent	Good	Excellent
with schedule visibility	Poor	Poor	Poor	Excellent	Excellent	Good	Good

Prior to the development of the proposed model, a user requirement was conducted to ascertain that the intended output of the model is streamlined with the end users' needs. In instances where these needs were unclear, it was difficult to understand them by talking about them and explaining them with written reports. It was, therefore more effective for the intended users to interact with the prototype to understand what the new system could do and how to best apply it to their needs. System prototyping, therefore is usually more appropriate when user requirements are unclear, because they provide prototypes for users to interact with early in the SDLC (Alan, Barbara & Roberta, 2012)

The technology used in this study was not entirely new and thus the risk involved was relatively low. Methodologies that are therefore applicable to high-risk development were therefore not applicable in this study. However, in as much as the technology being applied to this study was not entirely new, the proposed solution to the identified problem outlined in section 1.1 was intended to be met through the fusion of two pre-existing technologies namely artificial neural networks and dimensional data. The proposed solution, therefore, merits to be classified as a complex system which requires careful and detailed analysis and design. Though not very suitable for such scenarios, system prototyping development methodology was deemed acceptable for this study due to its flexibility when it came to short time schedules. System prototyping is acclaimed to be excellent choices when time lines are short because they best enable the project team to adjust the functionality in the system on the basis of a specific delivery

date (Alan, Barbara & Roberta, 2012). The general flow system prototyping is as illustrated in Figure 3.1

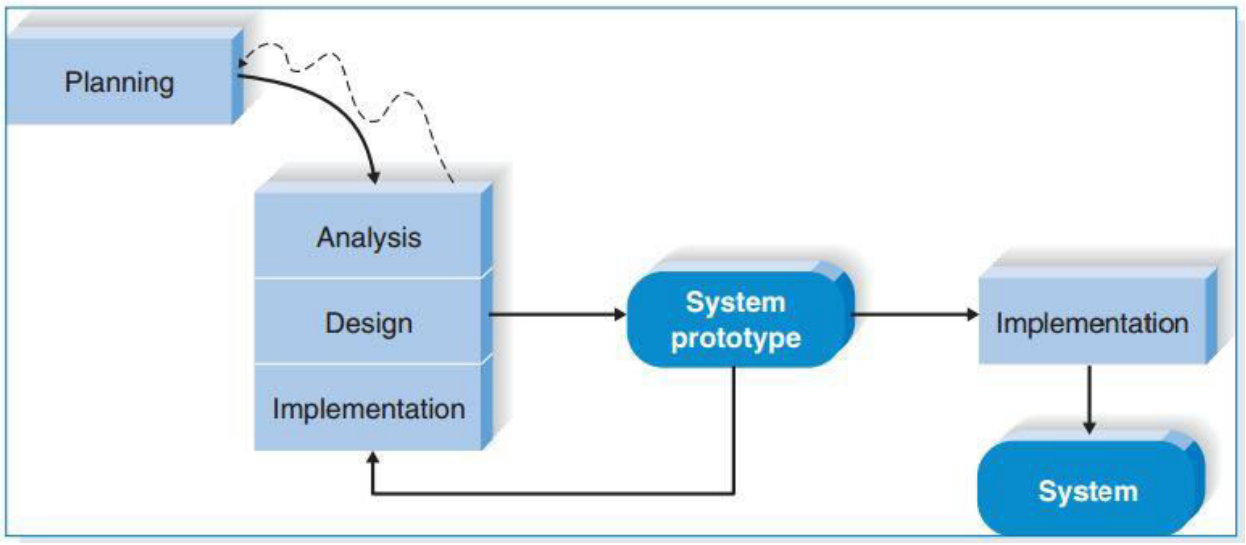


Figure 3.1: System Prototyping Development Methodology (Alan, Barbara & Roberta, 2012)

System prototyping works by performing the analysis, design, and implementation phases concurrently in order to quickly develop a simplified version of the proposed system and give it to the users for evaluation and feedback. Upon receiving feedback from the intended users of the system, the developers are tasked with the duties to reanalyze, redesign, and re-implement a second prototype that corrects deficiencies and adds more features. This cycle as outlined by Alan et al., (2012) continues until the analysts, users, and sponsors agree that the prototype provides enough functionality to be installed and used in the organization. The approach as further explained by McConnel (2003) is very useful when users have difficulty expressing requirements for the system.

3.3 System Analysis

Systems analysis as explained by Alan et al., (2012) is a process of collecting factual data, understand the processes involved, identifying problems and recommending feasible suggestions for improving the system functioning. This usually entails understanding the business processes; which in this case is the admission process, gathering operational data, understand the information flow, finding out bottlenecks and evolving solutions for overcoming

the weaknesses of the system so as to achieve the organizational goals. The system analysis process in this study in general, sought to answer the questions of: who used the proposed model and what model was used.

The first stage of the system analysis phase was to include a study of any pre-existing enrollment prediction approaches within the location of study. This was to aid the researcher to identify the existing loopholes in the current system and thus develop a solution. The next step was the requirements gathering which as earlier outlined in section 3.2 was a crucial stage in the system prototyping development methodology. Finally, a research proposal was compiled and delivered to the faculty.

3.4 Research Design

This research design approach followed was the experimental research. This approach as highlighted by Creswell (2003) works to the respect of first identifying the research objectives. The second step involves building a model as a proof of concept. The next stage encompasses sampling and an evaluation of external validity through data collection and data analysis that includes an evaluation of statistical conclusion validity. The results are then reported and a conclusion arrived at. This is as illustrated in Figure 3.2. This research design suited this research since data relating to students was collected and analyzed using the machine learning algorithm, the results were then validating to test for the accuracy of the model.

3.4.1 Location of Study

The system was deployed in Strathmore University, a private local university in Kenya. This University was chosen because of an existing data warehouse and up to state ETL platform. Additionally, following recent analysis on the enrollment data by the University data analysis unit, interesting patterns in the admissions dashboard made it a preferable study setting.

3.4.2 Target Population and Sampling

The unit of analysis in this study were students. Specifically, this study was keen on admitted students within Strathmore University and their transition to the next level of

institutional recruitment which is enrollment. The target population for secondary data for external validation (as explained in section 3.4) was classified into two. First the intended end users of the proposed model who are the staff in the admission department and the students who have been admitted (either enrolled or not enrolled). Random sampling was thus deemed to be most applicable in this case since it is of interest to this research to draw data from two different groups that bear different characteristics.

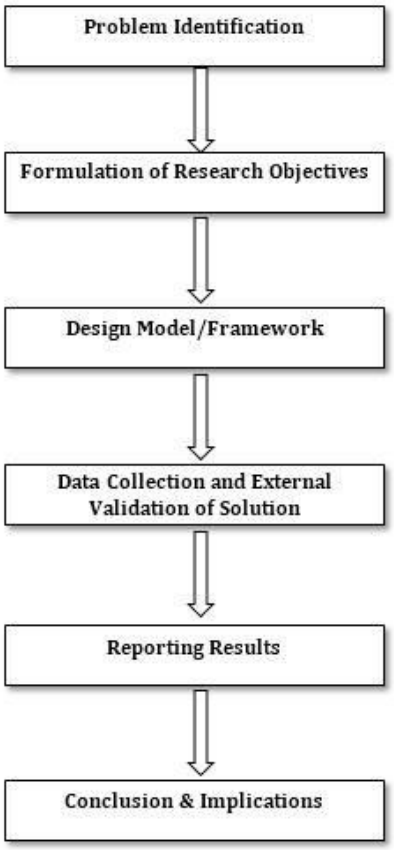


Figure 3.2: Experimental Research Design (Creswell, 2003)

According to Kenya National Bureau of Statistics (2015), the number of students who were enrolled in 2015 was 5012, which translated to about 60% of the total admitted in number. This thus makes the target population for these two strata to be 5012 for the enrolled students

and approximately 3314 for those who did not show up. According to Albandoz (2001), the main estimators of stratified random sampling are the total population as shown in the equation. Through the proportion estimator stratified random sampling, a sample of each student status was arrived at. Data was collected from staff within the admissions department due to the fact that they are few in number. The Equation 3.1 was used at arriving at the population size. The sample size thus constituted of 5 staff members. For the data used in training the model, 89916 student records were used.

$$\widehat{X} = N \sum_{i=1}^{n} \frac{X_i}{n}.$$

Equation 3.1: Random Sampling Proportion Estimator (Albandoz, 2001)

Where:

$\bar{X}_h$ is the sample average for variable X in stratus h.

N_h is the size of population h.

N is the size of the population.

n_h is the sample size in stratus h.

n is the sample size.

P_h is the sample proportion of the variable in stratus h

3.4.3 Data Collection and Procedure

Both primary and secondary sources of data were used in this study. The data used in training the model was collected from secondary sources such as existing databases. While conducting user acceptance, interviews were used for the reason that it made available the feedback in the shortest time possible. The use of interviews also helped in gaining more insight on the methods currently used in Strathmore University for enrollment forecasting as well as finding out the impact of not being able to predict enrollment rates. Other secondary sources that

were used in data collection were written reports and journals regarding existing systems or models or algorithms that are currently being employed for enrollment prediction by universities in Kenya.

3.4.4 Data Analysis

Since interviews were used to conduct user acceptance testing, the data collected was treated as being qualitative in nature. The aim of collecting data was to understand the functional requirements of the proposed model from the intended users. To analyze this data, a deductive approach was used. The conclusions drawn from the literature reviewed in section 2 was used as the framework for drawing conclusions from the data collected. The analysis was used to inform the researcher on how they accepted the model as well as its relevance to them; thus making it valid for the researcher to recommend the proposed model.

3.4.5 Research Quality

Validity and reliability are key aspects of all research. Meticulous attention to these two aspects can make the difference between good research and poor research (Golafshani, 2003). The data analyzed in this research was intended to be consistent over time and to be a true representation of the population. As such, an effective sampling procedure, which in this case is the stratified random sampling was used when choosing the number of the respondents to interview.

To validate the results drawn from this analysis, a comparison between the findings and the reality was conducted. Unusual responses that span to be outliers were eliminated. This form of validation is known as data triangulation. Data triangulation as proposed by Rahman et al. (2012), allows for the retrieving data from a number of different sources to inform or test one body of data that is reliable. Additionally, the accuracy of the prediction enrollment model was computed so as to get informed about the level of reliability of the proposed solution as discussed in section 3.8

3.5 System Analysis

System analysis is a crucial step in the SDLC since it clarifies how the system will operate in terms of the hardware, software, and network infrastructure that will be in place; the user interface, forms, and reports that were used; and the specific programs, databases, and files that were needed (Alan et al., 2012). System analysis in this study involved the analysis of how data flows through the model, how the users interact with the model, the design of the data warehouse, the sequence of activities through the model and the different classes implemented in the algorithm. This was important to this research as it aided in outlining how the different facets of the proposed model worked to solve the problem identified in section 1.1. This was necessary for validating the research as a technically oriented solution to the identified research problem.

3.5.1 Data Flow Diagrams

The proposed model handles requests made by a user such as: how many are likely to get enrolled, who is likely to get enrolled or not and what is common among the un-enrolled. Data flow diagrams were modeled to show these flows of data through the system for purposes of providing an overview of the system. As is outlined in the conceptual design in figure 2.5, the model was fed with data from two sources; the transactional databases and third party satisfactory index data which showed the university's ranking at different points in time. Aspects such as dimensionality were added to the data sets and a series of manipulation done to the data set before, during and after it going through the artificial neural network for prediction.

3.5.2 Use Case Diagrams

Users interacted with the proposed system so as to gain from the predictions output by the proposed enrollment prediction model. Being a system, there arose a need for different user access levels which implies different functionality and interactivity across the different user types. Therefore, to demonstrate this clearly, use case diagrams were used for the purpose of showing the sequence of actions that provides value to the users.

3.5.3 System Sequence Diagram and Collaboration Diagrams

System sequence diagrams were used so as to show the progression of events over a certain amount of time in relation to use cases within the enrollment prediction system which had different kinds of scenarios presented to the users. Additionally, being a sophisticated prediction system, collaboration diagrams were used to show how the different software objects interact to produce the desired prediction.

3.5.4 Class Diagrams

Different algorithms were written in an object oriented approach and as such had their functionality represented in classes that interacted with each other especially in this case where the output of one algorithm becomes the input of another. To represent this approach, class diagrams were used for the purposes of showing attributes, methods and the relationships among these objects.

3.5.5 Star Schema

One of the key features of the proposed system was the data warehouse from which it sourced the data it used for the machine learning algorithms. A data warehouse is of peculiar nature. Unlike the transactional databases that are represented using database schemes, the proposed data warehouse was represented and designed from a star schema that showed how the various dimensions and facts interacted, and how the student admission data, as well as the university's satisfactory index data, were dimensionally represented within the various fact tables and dimensions.

3.6 System Implementation

The system was developed iteratively as is the procedure used for system prototyping development methodology. The use of a data warehouse as a data source, which is served by an extraction, transformation and loading module was under constant development based on the nature and form of the data required by the machine learning algorithms. The machine learning

algorithms being self-learning, was repeatedly tuned on few occasions and thereafter left to self-adjust.

Changes to the proposed system regarding the ETL were anticipated based on the nature of data required by the machine learning algorithms and fixes in response to these changes were made.

3.7 System Testing

The proposed model was tested to ensure that it solves the research problem for which it was intended for. Due to the fact that the proposed model was mostly as an integration of the data warehouse ETL platform, the data pre-processing platform, and machine learning platform, a unit testing approach was followed to check for the proper functioning of each module. Additionally, accuracy tests were carried out to measure the level of enrollment prediction accuracy of the proposed model.

3.8 System Evaluation and Validation

System evaluation and validation involved testing for the accuracy of the proposed prediction model in relation to the reality to validate the accuracy of the algorithm. Additionally, the system was also subjected to a heavy amount of data that was very noisy, which is the case in most institutions of higher learning. This was important so as to establish how well the proposed model would handle this situation and at what speed.

Chapter 4 : System Design and Architecture

4.1 Introduction

This section describes in detail the general architecture as well as the comprehensive design of the proposed solution of the dimensional student enrollment prediction model. As outlined in section 2.6, the proposed model has a number of components and processes as well as users who interact with it. The design diagrams discussed in this section follow the Unified Modelling Language approach and include the use case diagram which details user interaction with the system, class diagram which shows how different class objects in the proposed model interact, system sequence diagram which illustrates the sequential flow of processes within the model, a data flow diagram showing how data flows within processes and a star schema to show the facts and dimensions within the data warehouse that are used for storage of the students' dimensional data.

4.2 Data Analysis

The research approach taken in this study was qualitative. Structured interviews were conducted on 5 respondents who are the intended users of the proposed model. This interview was conducted with the purposes of ascertaining the acceptability of the proposed model to the users, their requirements that would best suit as a solution to the existing enrollment prediction problem within their departments and efficiency of the provided user interface which runs the proposed model.

The requirements of the interviewed respondents are as discussed in section 4.2.1. Most remarks from the respondents revolved around ensuring the data fed into the algorithm is clean and relevant enough while some of the respondents emphasized on the need to automate the whole data cleanup process. All the respondents also emphasized the need of ensuring that enrollment prediction is carried out in both batch and individual scale.

As relates to the need of the proposed model, most respondents outlined the absence of a platform that helps in predicting the yield rate since they have previously relied on their on speculative experience based approach to forecast whether an admitted student

is most likely to show up in class. Comparing their experience to the classification proportion of the model, the respondents all agreed that the model classifies the students in reasonable proportions.

The system with which the users interact with was considered to mostly user friendly and fast enough to be used in a production kind of environment and in large scale. The accuracy levels were acceptable though it was taken into advisement that the accuracy of the model should improve with time.

4.3 Requirements Analysis

Following the research objectives outlined in section 1.3, a requirements analysis was conducted to outline the functional requirements that the model should meet, the usability requirements that is needed for the proposed model to be usable, reliability requirements and a supportability requirement of the proposed model.

4.3.1 Functional Requirements

From the structured interview conducted, the users' responses were analyzed and the following requirements deduced as the appropriate functional requirements for the proposed model.

i. The model should be able to extract highly normalized data from transactional databases, denormalize the data, encode the nominal values into numerical values then add a time variant attribute and satisfactory index data to the dataset extracted.
ii. The model should be able to load the dimensional data into the preliminary fact tables and dimensions within the data warehouse.
iii. The model should be able to scale the row values of the dataset, split the dataset and load the scaled data into desired workbooks or text files

iv. The model should be able to select suitable features that are necessary for student enrollment prediction and write the weightings of these features within a text file.
v. The model should be able to train itself based on user defined training parameters
vi. The model should be able to predict student enrollment prediction; either in batch student enrollment prediction form (showing number of students likely to get enrolled) or in the form of individual student prediction (showing whether or not a particular student is likely to be enrolled)
vii. The model upon request should be able to generate an accuracy report that validates the prediction model

4.3.2 Usability Requirements

The intended users of the proposed solution in this research are the admission staff within Strathmore University and the staff within the Registrar's office. Therefore, the interaction between these users and the model was made simple and very interactive to aid them in making student enrollment prediction through the platform an easy exercise. This was foremost guided through the design of friendly user interfaces with minimal work at the dispensation of the user.

4.3.3 Reliability Requirements

i. A regular snapshot capturing instances of the data at different times should be conducted and stored both in a local and server based data warehouse as a backup plan as well as for the sake of consistency in tracking any changes within the dimensional students' data
ii. Should the model fail to run, the administrator should be able to revert to the last stable pickle file that contains the prediction model and proceed to troubleshoot the model for the occurring issue(s).
iii. The model should have the ability to denormalize highly normalized datasets and add a dimensionality aspect through attaching of the of a time variant feature to the dataset
iv. The model should be able to predict student enrollment with a satisfactory degree of accuracy.

4.3.4 Supportability Requirements

The platform accessible to the users is a web based application and should thus be easily usable on desktops or laptops through any browser of user's preference. Navigability through the web application is made clear for a user friendly experience.

4.4 System Architecture

Figure 4.1 shows the general system architecture of the proposed model. The architecture is demarcated into two major parts, the user portal, which is the platform through which the user interacts with the model and the dimensional enrollment prediction portal, which is an artificial neural network based model that is used to predict student enrollment. Prior to interacting with the system, extraction, transformation and loading is conducted to pull the data from pre-existing academic transactional databases. Data pre-processing then takes place after which a pre-built artificial neural network is trained using the dimensional student data. It is after these process that a user is able to make contact with the model, make a prediction or accuracy report request and receive back a successful feedback in form of enrollment prediction or accuracy report from the model.

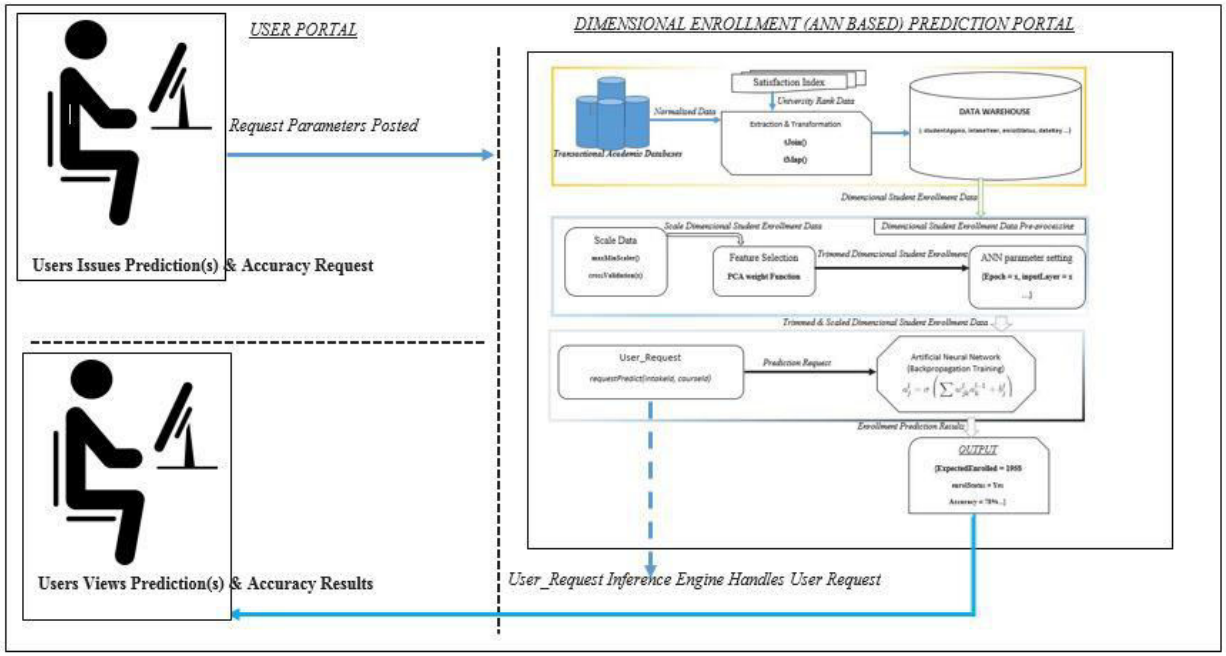


Figure 4.1: System Architecture

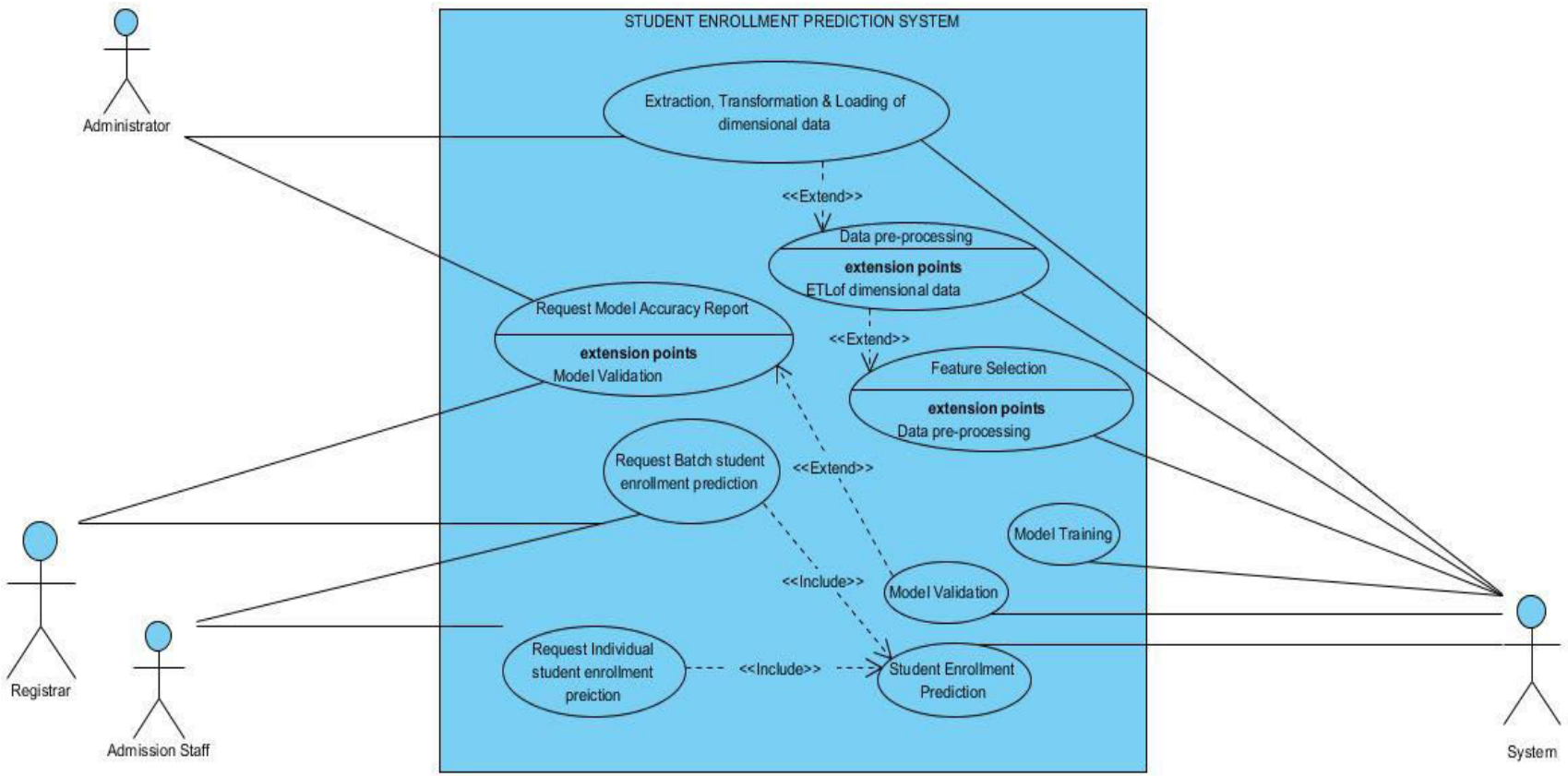


Figure 4.2: Use Case Diagram

Table 4.1: ETL, Pre-processing and Feature Selection

Use case: Perform Extraction, Transformation and Loading of Undimensional data, Pre-process Dimensional data, Perform Feature selection	
Primary Actors Administrator System	
Preconditions Working Data integration platform Data pre-processing library Feature selection inference engine	
Post conditions The data output is dimensional The data output is trimmed to have attributes necessary for predicting student enrollment	
Main Success Scenarios	
Actor Intention	**System Responsibility**
1. Administrator builds or rebuilds and schedules an extraction job	
2. Administrator develops Data transformation constructs	
	3. System runs Data extraction and transformation
	4. System loads dimensional data into the data warehouse
5. Administrator prepares feature selection constructs	
	6. System outputs weighted features relevant for student enrollment prediction

4.5 Use Case Diagram

The user interaction with the proposed model is illustrated in the use case diagram in Figure 4.2. The main actors as shown in the use case diagram are the administrator, the registrar, the admission staff and the system itself. First, the administrator interacts with is the extraction, transformation and loading of dimensional data. The Administrator is also able to request for accuracy report from the system. The generation of accuracy report relied on the model validation use case which is run and accessed by the system as an actor. The admission staff requests for either batch or individual student enrollment prediction both of which are reliant on the student enrollment prediction use case which are also accessed and facilitated by the system as a use case. Finally the actor requests batch student enrollment prediction as well as request for

an accuracy report. Table 4.1, Table F.1, Table F.2 and Table F.3 describe the use case diagram in detail outlining the main success scenarios for each use cases.

4.6 Data Flow Diagram

The Figure 4.3 illustrates the data flow diagram. This diagram shows the storage, the processes and the data flows that occur within the proposed system. The first process is the extraction, transformation and loading of data. The data source that informs this process is the normalized students' data store which holds students information in form of relations. A satisfactory index which holds a history of past university ranking is also fed into this process. This storage is sourced from reliable online university ranking portals. Upon the completion of this process, the dimensional data is then loaded into the unscaled data store which stores unscaled data of students which is dimensional. The process that follows is feature selection, which sources its data from the unscaled students' data store. This process is tasked with trimming the dataset by selecting the suitable features necessary for making enrollment prediction. The output of this process is stored in the trimmed students' data store.

The process that follows is the data pre-processing process which scales the data values from the trimmed students enrolled data store. The output is then stored in the pre-processed students' data store. The model training process then executes, with its data being sourced from the students' pre-processed data store and its output being stored in a prediction model data stores which in this case is not a database but pickle (.pkl python file) file which is executable. This stored model is then used in the subsequent processes of testing the model, validating the model, making individual predictions and making batch student enrollment predictions.

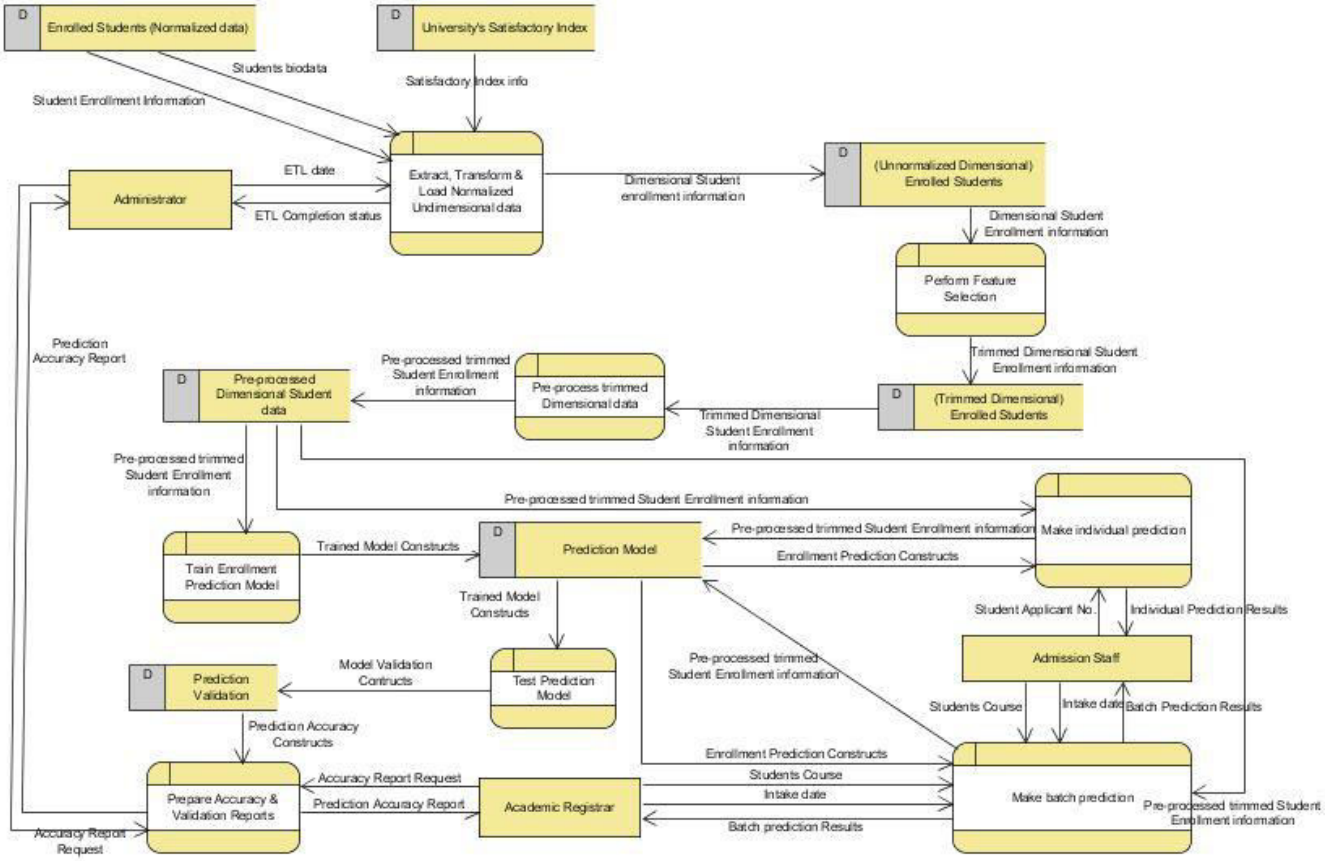


Figure 4.3: Data Flow Diagram

4.7 System Sequence Diagram

It is intended that the model shall support interactivity both between itself and the users as well as between its different components. Figure 4.4 shows the sequence diagram that illustrates the sequences of requests and feedbacks in and out of the system.

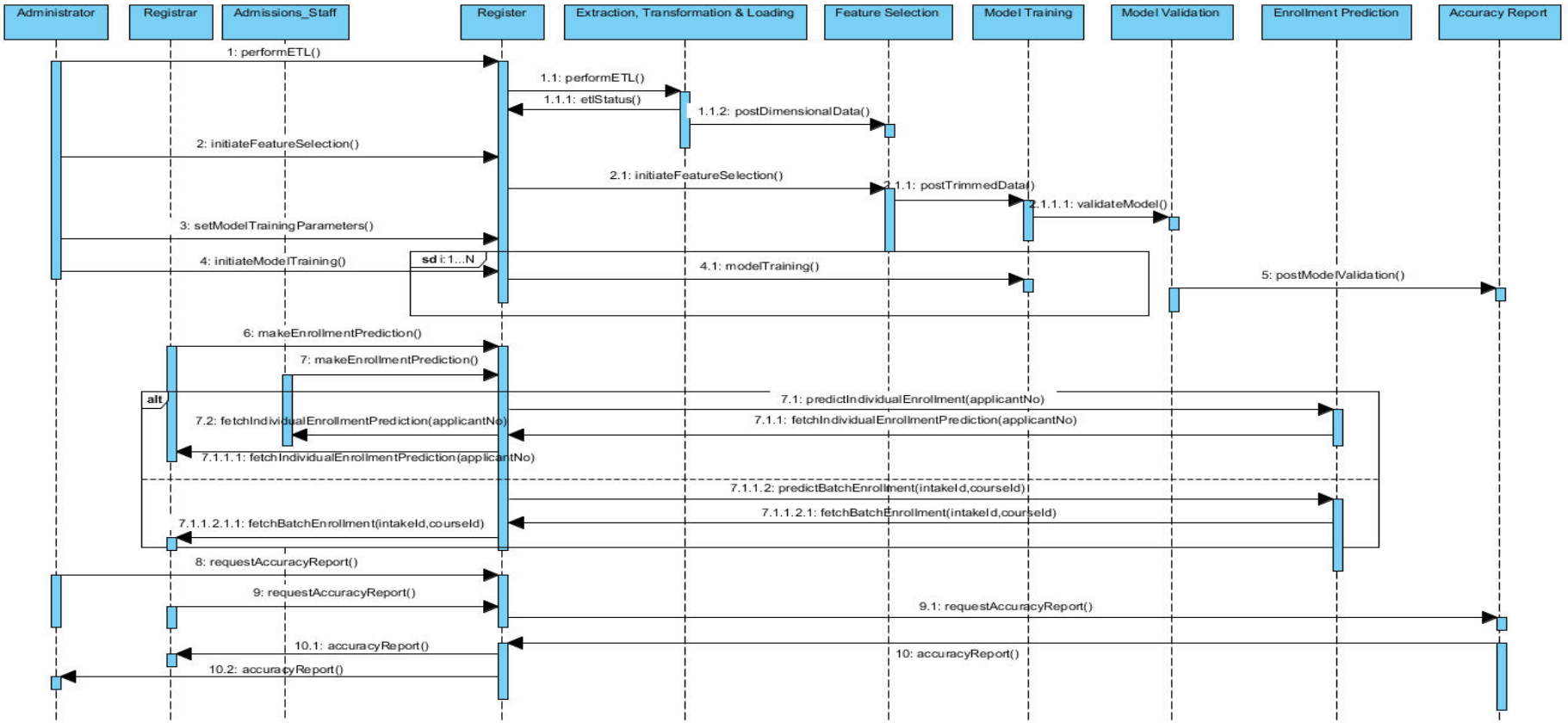


Figure 4.4: System Sequence Diagram

Being an extension of the use case diagram the sequence diagram in figure 4.4 also starts with the administrator issuing a command to initiate extraction, transformation and loading (ETL) process, the completion or failure of which results in a completion status response showing whether or not the ETL process was completed successfully. The Administrator is then able to initiate the feature selection process, which upon completion results into the posting of trimmed dimensional attributes to the model training process.

After the completion of the training process, the admission staff is able to request a student enrollment prediction. A student enrollment prediction can take two alternatives. First, the admission staff may wish to check whether a particular individual student who is already admitted is likely to get enrolled. This command is issued with the applicant number parameter, and returns a Boolean result showing whether or not the student is likely to be enrolled. The second alternative to this process is the batch student enrollment prediction.

The admission staff or the registrar may wish to know the numbers of students likely to be enrolled as batch. To do this, they issue a batch enrollment prediction command, whose parameters are the intake year and course of choice. The feedback of this command is the total number likely to be enrolled per course in a given intake period, as well as a detailed list of the individuals showing their likelihood of enrollment.

To get an accuracy report, both the administrator and registrar issue a request accuracy report. The feedback is posted to both the administrator and the registrar showing the accuracy of the model. The accuracy output here is generic, since it is based on the entire validation of the prediction; that is, even before running a prediction job, both the administrator and the registrar can request the validation results which are basically stored in the validation pickle file which is generated upon the completion of the model validation process.

4.8 Class Diagram

The proposed model shall follow an object oriented approach in the development of its logic. This implies that each component will be implemented in classes through well-defined functions. Figure 4.5 shows the proposed classes that shall be used in the development of the classes to be used in the proposed model and how they interact.

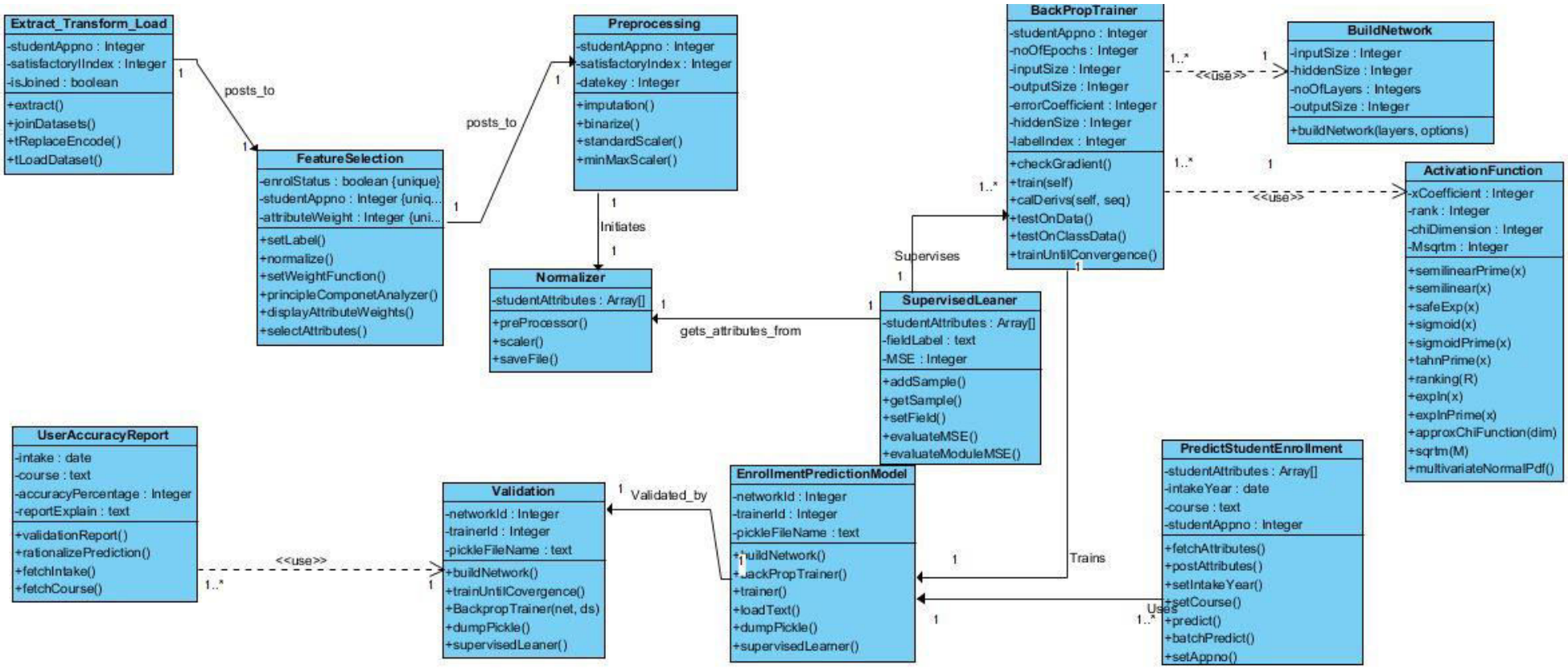


Figure 4.5: Class Diagram

The classes can be classified into three major categories. The first category are the classes that pre-process the data. These classes will include, Extraction Transformation and Loading class, which is a class that will be written in java language and is used majorly for data integration by pulling highly normalized data from transactional databases of pre-existing academic management systems, transforming them into dimensional data and loading them into the data warehouse. The other class in this category will be the pre-processing class which is a class written in python programming language and is used majorly for scaling the data and dividing the dataset into sub-datasets of training dataset, validation dataset and testing dataset. This class uses the scaling functionality defined in the normalizer class. The training dataset is then used in the other class of this category known as the feature selection class. This class will contains functions to be used in labelling the dataset by attaching the target attribute to the enrolled-status attribute. It then uses principle component analysis algorithm to attach weights the remaining attributes in the dataset, in rank of importance. These weights are then used to show which attributes are necessary for student enrollment prediction. The training is concluded with pickle file containing the prediction model. The Validation class is then used to validate the trained model.

The second category of classes is the model training category. This category foremost contains the *BuildNetwork* class which is used for building the artificial neural network used in this model. This class has as some of its attributes, the number of input nodes, the number of hidden nodes and the number of the output nodes and size of hidden layer. This class is used by the *BackPropagationTrainer* class which based on a specified number of epochs will train the model as supervised by the *SuervisedLearner* class that supervised the learning process by continuously pointing the attributes to the label field which in this called is the enrolled-status attribute. Error correction mechanisms will be used in adjusting the model during learning and such as the Tanh activation function contained in the *ActivationFunction* class shall be used. The model will keep on adjusting its weight as it checks the error co-efficient of itself.

The final set of classes to be used are those that the users shall interact with. These include *EnrollmentPredictionModel* class which interfaces both the *RequestAccuracy* class and *StudentEnrollmentPrediction* class. These classes shall aid the users to get student enrollment prediction reports as well as accuracy reports of the same as shown in Figure 4.5

4.9 Star Schema

The dimensional data that bears the student enrollment information stores the data in the data warehouse in form of one fact table and dimension tables as shown in the star schema in Figure 4.6

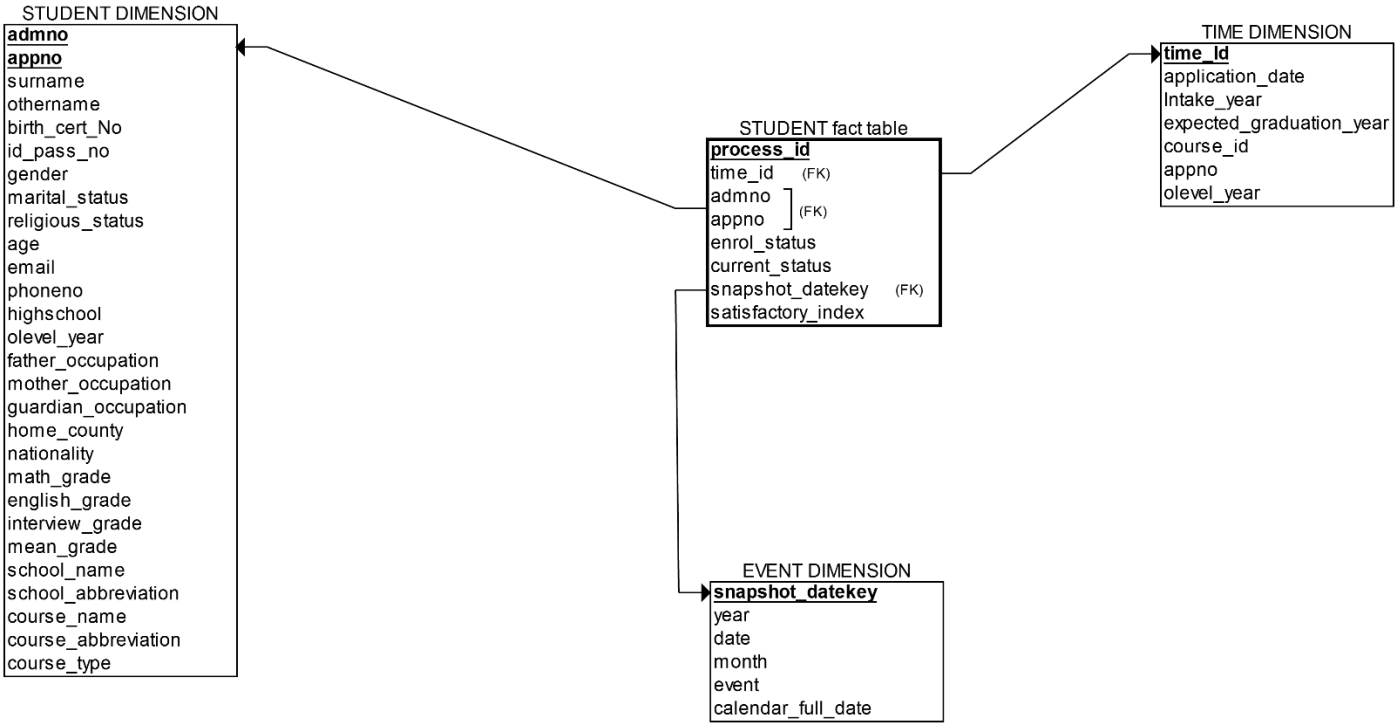


Figure 4.6: Star Schema

Unlike the conventional transactional database schemas, the star schema in Figure 4.6 shows how the dimensional data pertaining to student enrollment is stored in the data warehouse. The data is highly denormalized as seen by the possible repeating groups in the Student dimension. The dimensionality of the data is not only enforced through the denormalization but also through adding of the time variant attribute. This is possible through the time dimension showing the time details of the enrolled students and the event dimension that bear the details of when the snapshot was taken in relation to any prevalent events.

Chapter 5 : Implementation and Testing

5.1 Introduction

The implementation of the proposed prototype is discussed in this section as well as the testing process of the proposed prototype. The implementation of the model followed after four activities. The first step, prior to the development of the model, was to conduct the feature selection so as to ascertain the attributes that have a degree of co-relation to the target class; which is the enrollment status. The second was data extraction which entailed extracting data from the various existing transactional databases and transforming them into dimensional data. This transformation involved adding a time variant to the different snapshots of the data taken relating to both the student data and third party satisfactory index data. Data pre-processing was then conducted by encoding the nominal data to numeric data and loading the data into the data warehouse. The data from the data warehouse was then normalized and split into training, testing and validation set. The training set was then fed into the neural network model which shall be discussed in the sections to follow.

5.2 Model Components

The proposed model is made of two major parts, the data pre-processing part, and the machine part. They are as discussed in the sub-sections that follow.

5.2.1 Dimensional Data Extraction Transformation and Loading Component

Data used in the model originally exists in transactional databases and flat file storage. The Neural Network only accepts numerical values for purposes of formulating an accurate prediction model. The data, however, exists in nominal form. The data thus first had to be extracted from these transactional databases and denormalized. This included performing left joins on highly normalized tables and mapping the resultant relation into a large table known as a fact table. The original student data used in this study exists in an oracle database. Upon extraction, the resultant dimension had to be placed in a staging area awaiting a time variant attribute known as datekey to be appended to it. The data extraction, transformation, and loading were necessitated through Talend Open Studio for Data Integration Tool as shown in Figure 5.1

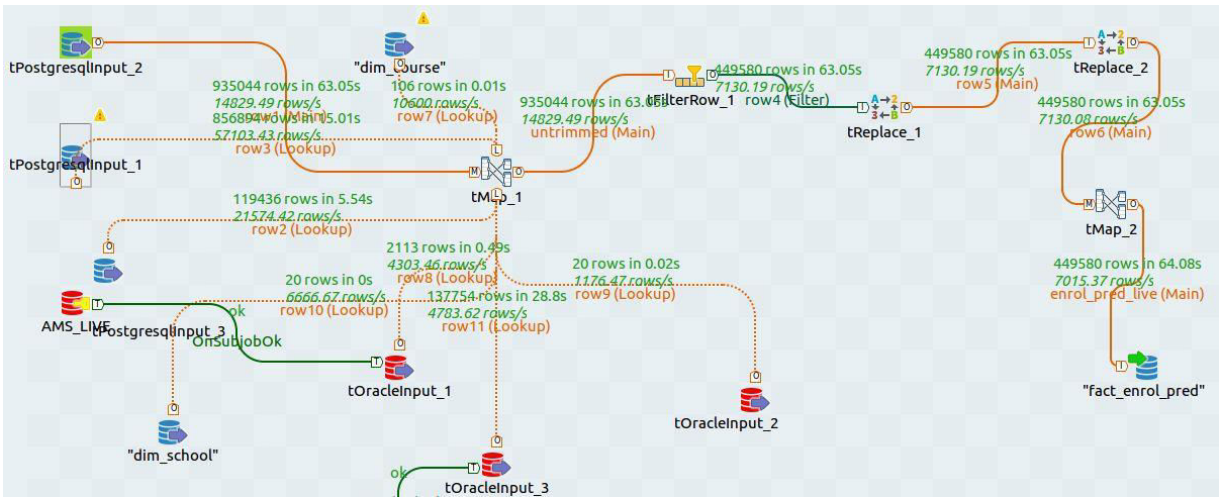


Figure 5.1: Staging Area for Transactional Data

The *datekey* attribute stems from an event dimension which is a calendar in the form of relation detailing the day and any event that may have occurred on that given day. The assumption made here is that the datekey attribute caters for any other external factor that may lead to enrollment such as ongoing admissions in neighboring Universities or even an ongoing nationwide financial crisis that could directly impair one's enrollment process. These factors are directly linked to the calendar since they normally follow a uniform fiscal calendar. Figure 5.2 shows a snippet of the event dimension as exists in the prototype's data warehouse.

PostgreSQL 9.4.10 running on [redacted] -- You are logged in as user "alaka"

phpPgAdmin: Live?: sudw?: public?: dim_event?:

Browse

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 Next >

Date	Full_Day_Description	Day_Of_Week	CalendarMonth	calendaryear	Fiscal_Year_Month	event	Weekday Indicator	datekey
2012-01-01	January 01, 2012	Sunday	January	2012	F2012-01		Weekend	1828
2012-01-02	January 02, 2012	Monday	January	2012	F2012-01		Weekday	1829
2012-01-03	January 03, 2012	Tuesday	January	2012	F2012-01		Weekday	1830
2012-01-04	January 04, 2012	Wednesday	January	2012	F2012-01		Weekday	1831
2012-01-05	January 05, 2012	Thursday	January	2012	F2012-01		Weekday	1832
2012-01-06	January 06, 2012	Friday	January	2012	F2012-01		Weekday	1833

Figure 5.2: Snippet of Event Dimension

5.2.2 Feature Selection

This component entailed carried out feature selection so as to ascertain which attributes would best fit in trying to predict whether a given student will be enrolled. Various approaches were used and compared in order to arrive at a more informed conclusion of the features required. Ensemble feature importance, Principle Component analysis and Recursive Feature Elimination methods were used and comparatively applied in arriving at the most suitable attributes which are discussed in Table 5.2. In principle component analysis the desired attributes alongside the target class were input into the principle component algorithm, and the correlation between the attributes and the target class computed. The output of the principle component analysis were principle components which showed the weights given to each attribute being their correlation to the target class. Likewise, the Ensemble feature importance also gave weights of correlation to each attribute. Recursive feature elimination on the other hand ranked the attributes in order of importance while eliminate those found to have minimal correlation to the target class.

Table 5.1: Feature Selection Results for Multiple Algorithms

Attribute	Ensemble Feature Importance (Weight)	Principle Component Analysis (Weight)	Recursive Feature Elimination (Rank)
Expected Graduation Year	0.02	0.05833107	3
Application Date	0.007	0.01330968	9
Course	0.02	0.22539487	4
Program Type	0.07	0.13509157	6
Study Mode	0.19	0.00264996	2
School	0.09	0.22469152	1
Datekey	0.05	0.00406003	11
Gender	0.01	0.03923693	13
Marital Status	0.03	0.20332033	12
Age	0.01	0.17807036	8
Nationality	0	0	15
Religion	0.003	0.08372524	14
Mean Grade	0.05	0.50392839	10
Math Grade	0.03	0.51102596	5
English Grade	0.01	0.51388177	1
High School	0.02	0.0435967	7
Satisfaction Index	0.8	0.11913779	1

As shown in Table 5.1, the nationality attribute bore no co-relation to the target and even ranked the last in the Recursive Feature Elimination process. The Satisfaction index attribute bore most of the weight in terms of co-relation to the target class. Other attributes that were also significant but ranked low in the feature selection include the datekey attribute which to the contrary is necessary for adding the dimensionality to the dataset.

5.2.3 Data Pre-processing Components

The first step of data pre-processing starts was the staging area where the nominal values were encoded and thus transformed to numerical values. Binary-based attributes for instance gender were given values binary numerical values such as such 1 for male and -1 for female. Values with a wide range of instances such as grade marks were given values within appropriate ranges with grade 'A' being labeled as 12 and the least grade which is F being labeled as 1. The null values were labeled as -1 while target class which is the enrollment status was labeled as 1 for enrolled students and -1 for students who are not enrolled as shown in Figure 5.3

InputColumn	Search	Replace with	☑ Whole word
app_status	"Enrolled"	"1"	☑
app_status	null	"-1"	☑
olevel_type	"0"	"-1"	☑
olevel_type	"2008"	"1"	☑
olevel_type	null	"-1"	☑

Figure 5.3: Data Encoding Snippet

The second step of data pre-processing entailed fetching the data from the data warehouse and splitting the data into training, validation and testing sets. These sets were then stored in delimited text files in preparation to be used in the neural network. The reason for choosing the text delimited files was influenced by the flexibility of the programming language used, which offers a variety of options for manipulating delimited texts files into arrays, which are then more easily normalized as discussed in the normalization section.

5.2.4 Neural Network Components

An artificial neural network was built to train the machine learning algorithm for the proposed prototype. The Scikit-Learn python library was used and modified appropriately to handle the dimensional data used in the model. Other data pre-processing classes were added to

aid in improving the accuracy of the model. The general layout of the neural consisted of an input layer, hidden layers, and the output layer.

5.2.4.1 Input Layer

The hidden layer consisted of 15 nodes which were shaped to be equal to the size of the attributes chosen to be used in the model. These attributes included the student's; expected year of graduation; which is a derived attribute from the intake year and the program type taken by the student. The students application date, course, program type, study mode, school or faculty, datekey, gender, marital status, age, mean grade, math grade, English grade and the high school attended, also included the data that was used in training the neural network. Additionally, the satisfaction index was also added to the dataset following high ranking in the feature selection as a factor that influences student enrollment. The satisfaction index is the university ranking at the moment that the student was being admitted.

5.2.4.2 Hidden Layer

The model consists of two layers each with 15 nodes that pick from the input layer of the neural network. The hidden layer employed a feed forward approach known as a multilayer perceptron (MLP) which uses a supervised learning technique known as backpropagation. This technique works by observing the target class and adjusting the connection weights between the nodes so as to arrive at the most acceptable set of weights necessary for predicting whether or not a student would be enrolled.

The activation function used in the hidden layer was Tanh as opposed to the sigmoid activation function. The reason for using Tanh is the target class was encoded as -1 for students who were not enrolled and 1 for students who are enrolled. Additionally, the use of -1 in encoding missing values necessitated the use of Tanh activation function. Tanh activation function which is also known as the hyperbolic tangent function is an antisymmetric activation function that permits and scales well for values in the range [-1, 1]. In as much as Tanh is a rescaling of the sigmoid function, the accuracy of the model improved from 68% accuracy to over 71% accuracy when the Tanh activation function was used instead of the sigmoid activation function. Equation 5.1 shows the Tanh activation function.

$$\text{Tanh}(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$ Equation 5.1: Tanh Activation Function

The solver parameter implemented in the hidden layer is known as Adam. Adam is a derivative of the stochastic gradient descent learner. This algorithm came in as suitable since is it was simple to implement and use little memory requirements. Additionally, it was undistracted by scaling of the values used in the chosen features which were also noisy as they contained lots of null values.

5.2.4.3 Output Layer

The enrollment status is the target class for the model. The output layer thus consisted of two nodes that took the size of the output target size. Values entered into the neural network were thus, in the end, classified as either 1 for enrolled students or -1 for students who are predicted as not to be enrolled.

5.3 Model Implementation

Upon building of the neural network, the proposed model was implemented through first fetching the dimensional data from the data warehouse, then normalizing before passing it through the *StandardScaler* algorithm and feeding the scaled data into the neural network to be trained. The resultant was then stored awaiting execution.

5.3.1 Data fetching

A script was written for the model that fetches the string values of the dimensional data from the data warehouse and converting them to float values in preparation for normalization. This module then fits the data into arrays and splits the large data set into training, validation and testing sets which are then written into respective delimited text files. The splitting criteria had 70% of the total dataset being used for training, 20% being used for validation and 10% being used for testing.

5.3.2 Dataset Description

Following a detailed feature selection process a list of attributes was chosen to be used in the enrollment prediction model. The attributes within these datasets as well as their possible values are as detailed in Table 5.2. The possible values illustrate the nominal values as well as the representative numerical values that were used for training the proposed model. For purposed of coming up with cleaner data, only data relating to admitted full time undergraduates ranging the year 2013 onwards were used to train, validate and test the model.

5.3.3 Data Normalization

Prior to arriving at the most suitable data normalizing function, a variety of data scaling functions were tested. The first one to be tested was the *MinimumMaximumScaler*. This normalization function initially suited the model but crashed while trying to scale tuples with some values that are zero. This normalization function also performed poorly while handling columns that do not have a variety of values since it scaled them to a default of zero which was not suitable enough for the model.

The normalization function that was thus resorted to is the *StandardScaler* which unlike the Normalizer scaling function scaled the rows to a non-random range of values between -1 and 1. This normalization function also performed well in tuning the model to produce a better accuracy unlike the other normalization functions.

5.3.4 Model Training

The training of the proposed prototype was conducted in the hidden layer of the neural network. The training parameters were altered to suit the dimensionality of the data and favour outliers as well as the dirty data, especially the records that had lots of missing values. The first parameter that was altered was the learning rate which was reduced to 0.0005 since it was noted that having a higher rate and a low momentum led to non-convergence by the time the number of specified epochs was reached by the training algorithm. The numbers of epochs used in training was 2000 epochs.

Table 5.2: Dataset Description

Variable	Description	Possible Values
Expected Graduation Year	This is a composite attribute that tells when a given student is expected to graduate	Values greater than 2016
Application Date	This attribute shows the date when the admitted student applied for the course of choice	Values greater than 2012
Course	This attribute shows the course taken by the admitted student	BBIT(6), BCOM(7), MSIT(11) among others
Program Type	This attribute details the course type of the course chosen by the admitted student	Undergraduate(1), Postgraduate(2), Diploma(3) and Certificate(4)
Study Mode	This attribute shows the mode which the admitted student has chosen	Fulltime(1), Evening(-1) or Modula2(2)
School	This attribute gives the faculty that offers the course that the student so desires	FIT(1), SMC(2), SBS(3) among others
Datekey	This is the time variant attribute that bears the key when the snapshot was taken. It points to the event dimension in the data warehouse from which the training data set is sourced from	Values greater than 0
Marital Status	This attribute shows whether the admitted student is married or single	Married(1) or Single(-1)
Mean Grade	This attribute lists the grade that was scored by the admitted in high school prior admissions	A(12), A-(11), B+(10), B(9), B-(8), C+(7), C(6), C-(5) D+(4), D(3), D-(2), E(1), F(-1)
Math Grade	This attribute shows the grade scored by the student in mathematics prior admissions	A(12), A-(11), B+(10), B(9), B-(8), C+(7), C(6), C-(5) D+(4), D(3), D-(2), E(1), F(-1)
High School	This attribute shows the high school in which the admitted student was in	Values greater than zero (Primary key for high school relation used as encoding factor)
Satisfaction Index	This attribute is an aggregate rank of the university upon the time that the admitted	Value from -1(missing) onwards

	student was applying to the university	

This number was arrived following the observation that a greater number of epochs than 2000 started yielding a high relative mean squared error as shown. As such the momentum applied was high at 0.9. The Tanh activation function was used and Adam solver used. The dataset size used in the training of the model consisted of over 200,000 rows, sourced from different dimensionalities by virtue of cumulative snapshots taken from the transactional database.

5.3.5 Storing of Model

The trained model was then stored in an executable file that bears a pickle (.pkl) extension. This file type was chosen since it is well suited for storing models and can be executed easily in Python programming language. Additionally, the memory used while fetching the constructs of the file is quite as compared to alternative file type. This file named as enrol_pred.pkl, was then used while in the subsequent processes while validating and testing the model.

After validation, the model was then tested using unlabeled data in an attempt to classify admitted students. The fetching of this unlabeled data was achieved through the use of an interactive user interface tailored to be flexible for the end user as discussed in section 5.4

5.4 Software Flow

A web application to be used by the intended users was developed. This platform was designed and broken down into three simple steps as shown in Appendix D. The user first sets the criteria for the set of students for which they desire to conduct an enrollment prediction on. This is made possible through drop downs. The first drop down is used to specify the intake period while the second drop down is used to specify a course of choice. The second step that the user is required to do is to initiate the enrollment prediction by clicking on the Initiate Prediction button. The final step is the viewing of the prediction results which allows the user to download a workbook with the student's admission and their predictions results; which is a column that shows whether or not the given student is likely to get enrolled.

The above mentioned steps in the front end spun out a series of process in the systems logic. First the setting of the enrollment prediction criteria ends by the system writing the intake year and the course identification number to a data file known as *criteria.dat*. It is this data file that is read and used by the model to select the desired dataset based on the criteria specified in the data file. The second step which requires the user to initiate the prediction starts the normalization of the specified dataset. This process, though initiated in the frontend PHP platform, is executed in the *normalizer* python script.

Interactivity between the frontend PHP platform and Python business logic was made possible through the periodic reading and writing from stub files such as the *criteria.dat* file. Queries executed on both ends on a given session read from these stub files at interchangeable moments for consistency. The python scripts were on the other hand made executable and placed in the root folder where the PHP files exist to grant the service permissions to execute them. The resultant interface shown to the user is a simple web page that shows the total number of admitted students, the number predicted to be enrolled and the predicted number that are likely not to be enrolled. An expected enrollment rate (institutional yield) is also computed and displayed to the user.

Additionally, a brief accuracy summary is also presented to the user on the interface first in form of a confusion matrix then broken down into a table to show the model's accuracy. A button that allows for downloading of the prediction results data set is also provided for the user if they wish to view the exact details of the enrolled student.

The major challenge of following this approach as opposed to passing JSON encoded files between these two environments was the occasion storing of cookies which required the user to clear the cache for a consistent flow of data between these two environments. The development environment for the application is:

Linux Ubuntu 16.04 operating system
Codeigniter PHP framework
Postgres database management system
Python Programming Language
Talend Open Studio for Data Integration

5.5 Model Architecture

Figure 5.5 illustrates the proposed model for the dimensional enrollment prediction. The model's initial steps are data pre-processing where the data in highly normalized transactional databases are first denormalized and encoded into numerical data. Satisfactory index data sourced from reliable university ranking sites is also added to the dimensional data which is then normalized to scale well for the neural network algorithm. The training set is then passed through the multilayer perceptron and trained. The final is then written to an executable file known as enroll_pred.pkl which is called every time a user needs to make prediction on a given data preferable data set.

When either an admissions' staff or the University registrar needs to predict the expected enrollment status of a set of admitted, they first specify a criteria which selects the already encoded dataset which is stored separately in the data warehouse from those used for training and validating the enrollment prediction model for purposes of avoiding over-fitting of the model. The data that fits the user's criteria, which is in the bounds of a given intake period and a course. The data set is then scaled and executed in the enroll_pred.pkl which writes the results back to the date warehouse which is queried and shown to the user.

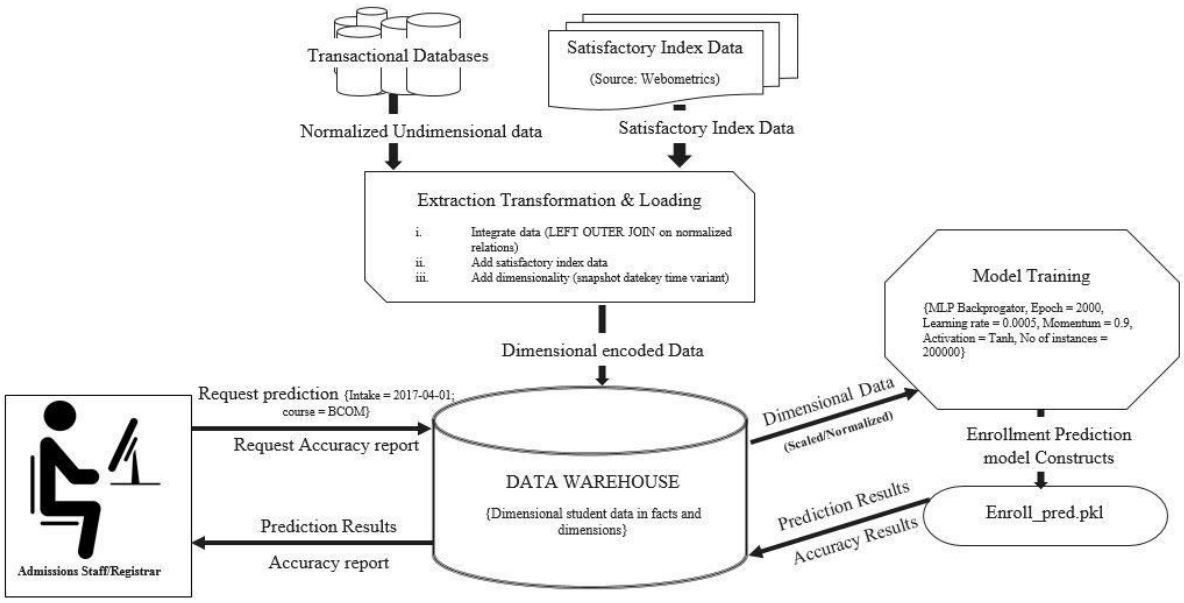


Figure 5.5: Proposed Model Architecture

5.6 Model Testing

Activities in this process involved testing the prototype for the required functionality, reliability, integration between each components and load testing. The priority for the functionality test is also outlined against each test case as detailed in Table 5.3.

5.7 System Testing

Activities in this process involved testing the proposed prototype for the required functionality, reliability, integration between each components and performance for the system to be used directly with the user. Priority relating to each system test is also outlined as illustrated in Table F.4.

Table 5.3: Model Testing

Test Case	Check Criteria	Priority
Reliability	Does the model consistently classify admitted students within the validated accuracy	High
Integration	Does the model integrate well the system used by	High

	the end users	
Load testing	How fast is the model while training a new data set that is large	Medium
Functionality	Does the stored model execute when required without crashing or malfunctioning	High
Functionality	Does data used in the model contain dimensionality in every snapshot	High

5.8 Model and System Testing Result

The proposed model performed successfully in predicting the enrollment status of admitted students and classifying them into two classes as either enrolled or not enrolled. The training of the model took a relatively shorter time considering the number of epochs used during training and also the number of instances used. Additionally the model integrated well the system to be used by the user. The system on the other also ran successfully and update the stub files accordingly in relation to the criteria set by the user at the given moment in time. Table F.5 shows the testing results for both the model and system.

5.9 User Acceptance Testing

The activities in this process entailed checking whether the system meets the specific objectives for which is was built for. This was done through handing the prototype to the user who then gave a feedback on the usability and reliability of the model and system that they used to interact with the model. Table F.6 shows test cases that were handled. The accuracy of the model was used to test how efficient the model was with higher accuracy being an indicator of high efficiency of the proposed model.

Chapter 6 : Discussions

6.1 Introduction

Following the analysis of the data gathered through an interview conducted by the intended users of the model within Strathmore University it was noted that there exists no standard method of forecasting the number of students to be enrolled. The users thus acknowledged that the use of the proposed model would seek to reduce the instances of poor decisions that have arisen through the old speculative approach employed by the administrative staff in different faculties as well as the admissions' department staff.

The accuracy level was satisfactory enough for the staff and was acclaimed as reliable enough for actual use in determining the number as well as the set of students likely to show up. Additionally, a detailed listing of these predictions as provided in the designed user interface of the model was also seen as useful as it would help in identifying the student who is likely not to get enrolled. This as outlined from the feedback gathered from the interview was analyzed to be very useful for other activities such as following up the individual students who fell under the not-enrolled cluster.

A review of models used previously for enrollment prediction was conducted and based on the nature of this research, the implementation of the dimensional student enrollment model was achieved through the development of an artificial neural network. This neural network was trained using dimensional student data. This model was then validated using already labeled data to test for accuracy of classification. The model's accuracy of the classification made it easier for the users of the system to consider it more suitable than the approach used by the admissions department staff initially. The approach that has for long been used by the admissions staff has normally followed a speculative based approach where the admissions staff or the different school administrators try to estimate the number of students likely to show up in the forthcoming intakes. This approach by virtue of being reliant on the human judgment has been biased and thus very unreliable.

The accuracy of the proposed model is attributed to the fact that the student data was transformed into dimensional by the addition of the time variant attribute. The use of this dimensional data within an artificial neural network yielded the high results as compared to when

the data lacked the time variant attribute. The use of a web application user interface enabled ease of use of the proposed model by allowing the user to make predictions on future intakes by simply following three simple steps. The speed of the web application in executing large sizes of predictions in a short period of time was made possible through the use of stub files that allowed both the web application and the data pre-processing scripts that are written in Python programming language to write on and read the stub files almost instantaneously. This feature also as gathered from the interview conducted by the researcher on the intended end users made the both the model and system acceptable since it is user-friendly and less tasking to the intended user.

6.2 Model Validation

To validate the model, error rate, precision, accuracy and recall ratio were used. The number of instances used for validation were 89916 which were sourced from the existing transactional databases that store details of admitted students. 64189 Instances out of 89916 instances were classified correctly while 25727 instances were misclassified. This thus resulted into an accuracy of 71.39% with an error rate of as detailed in Table 6.1

Table 6.1: Classification Output

Total Instances = 89916	No.	
Correctly classified instances	64189	(TP rate) 71.39%
Number of misclassified instances	25727	(FP rate)28.61%

To evaluate the performance of the proposed model, different evaluation classes were used and the results arrived as discussed in section 6.2.1

6.2.1 Detailed Accuracy

Table 6.2: Detailed Accuracy

	Precision	Recall	F1-score	Support	Class
	0.75	0.63	0.69	44457	Not Enrolled
	0.69	0.80	0.74	45459	Enrolled
Average/ Total	0.72	0.71	0.71	89916	

TP rate also known as sensitivity refers to the percentage of positive values that were correctly classified FP rate also known as specificity refers to the proportion of negative values in that were correctly classified. The results for sensitivity and specificity are summarized in the recall column in Table 6.2. The relevance of values used for classification was at 75% for the not enrolled class and 69% for the enrolled class; these two values indicate the precision of the model. To validate the performance of the proposed model, a receiver operator characteristics (ROC) curve was plotted. This is a graph that plots sensitivity against specificity to find whether the classification of the two classes, enrolled and not enrolled were separable as well as generally classified correctly. The area under the curve (AUC) shows the performance threshold for the proposed model. The output for this graph illustrated in Figure 6.1

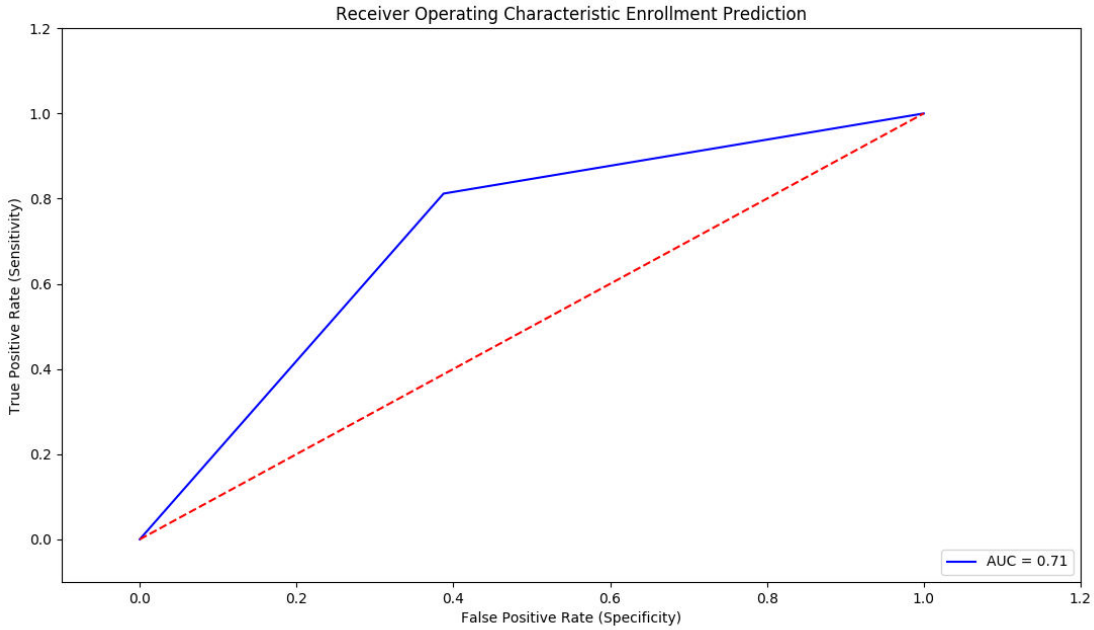


Figure 6.1: ROC Curve for the Proposed Model

6.2.2 Confusion Matrix

Validation for the student enrollment prediction was conducted after the training process. 20% of the total data set extracted from the data warehouse that contains the dimensional data. For purposes of visualizing the validation, a fact table listing the real values against the predicted values was populated and queried in order to find the numbers that were classified correctly as

either truly positive or truly negative. To arrive at the accuracy level of the model, the Equation 6.1 was used while to arrive at the error rate the Equation 6.2 was used.

$$\frac{tp+tn}{N} x\ 100$$ Equation 6.1: Confusion Matrix Accuracy Equation

$$\frac{fp+fn}{N} x\ 100$$ Equation 6.2: Confusion Matrix Error Rate Equation

This equation works by summing up the values classified as truly positive to the values classified as truly negative and the sum divided by the total number of instances used in the validation set. This results of are as summarized in the confusion matrix in Table 6.3. The accuracy of the model thus stood at 71.39% with an error rate of 28.61%.

Table 6.3: Confusion Matrix

N = 89916	Predicted: NO	Predicted: YES	TOTAL
Actual: NO	TN: 28033	FP: 16424	44457
Actual: YES	FN: 9303	TP: 36156	45459
TOTAL	37336	52580	

6.3 Contributions of the Model to Research

Following the gap that has arisen in Universities within Kenya, the proposed model seeks to solve the uncertainty when it comes to predicting whether an admitted student is most likely to transition into being enrolled. Through the proposed model, it shall therefore be possible, upon admission to tell the individual students who are likely to not get enrolled and thus follow them up. More importantly for the purposes of planning and budgeting, it shall be possible through the proposed model to know the expected institutional yield, which is the expected rate of turnover from the admitted students.

Additionally, the proposed model has proven that dimensional data performs well as the input for the artificial neural network since is shows the transition of different instances of snapshots of the dataset at different points in time. This dimensionality as discussed in section 5 is treated as a variation in the tuples which is suitable for training a neural classification model.

6.4 Shortfalls of the Model

The proposed model is limited to dimensional data. Therefore data from highly normalized relations existing in daily transactional databases would not scale or perform well with the proposed algorithm.

Additionally, the proposed model is a solution that solves a classification problem but not clustering or regression problem. As such, through the proposed mode, it is only possible to tell whether a student or a group of students will get enrolled or not. It is not within the scope of the proposed model to rationalize the classification results by listing the common attributes that the enrolled or not enrolled groups of students share or the exact attribute that could be causing them not to get enrolled.

Finally, the proposed is highly dependent on highly pre-processed data. As such, running it continuously could be quite tasking since the data has to be thoroughly pre-processed, scaled and transformed into dimensional data before being run through the model. At the moment, an automated Cron job reliant on the Linux operating systems seems to solve this issue, though this is not a permanent solution since for the prototype to be adopted into production scale, data pre-processing as a necessary step would need to be automated in any other operating system.

Chapter 7 : Conclusions and Recommendations

7.1 Conclusion

University admission in Kenya has relatively increased over the past years. However, the institutional yield which is the rate that the admitted students get enrolled has continued to be unstable and in some cases declined. Personnel charged with the responsibility of recruiting new students into these Universities are faced with uncertainty on the number of admitted students who might actually show up in class.

The admissions' department staff and the registrars within these Universities have resorted to relying on their experience, to make a speculative judgment on the number of students who are likely to get enrolled. This unreliable approach has resulted in short term solutions that are meant to correct the unexpected yield. In most instances, the Universities' administration has resorted to increasing the number of intakes as a cover up solution. This, in the long run, has compromised on the quality of the institution in terms of the quality of students being enrolled (Henriques, 2015).

This research therefore aimed at first identifying the factors that influence student enrollment in Universities and reviewing the current model that are used for enrollment prediction. A solution was then proposed in order to fill this gap by developing an enrollment prediction model that relies on dimensional student data. The proposed solution was tested for efficiency through validation that showed the accuracy of the model as being about 71% accurate. The dimensionality of the data was emphasized on due to the fact there are external factors mostly attributed to time that may affect the number of students likely to get enrolled. One of such instances includes ongoing admissions in other Universities. These events are captured in the event dimension and a time variant primary from this dimension is used in the data for training the neural network used for classifying admitted students as either enrolled or not enrolled.

In order to make the model easy to use for the intended user, a web application prototype was developed and tested by the intended users. The web application and the model which is packed in a pickle executable file operate in different environments and thus interact through parsing of objects through stub files which makes the interaction faster as acknowledged by the

intended users who tested the web application. The model upon validation was accurate as it classified most of the admitted students' instances correctly as discussed in section 6.2. The model, however, had some shortfalls as it is only limited to classifying the admitted student. The model does not seek to identify the reason why the admitted students were not enrolled by outlining the common features of the students who are predicted to most likely not to get enrolled.

7.2 Recommendations

In consideration of the results obtained from the implementation and testing of the proposed prototype, the following recommendations were made:

i. To reduce the levels of uncertainty as pertains to knowing the number of admitted students likely to get enrolled, an enrollment prediction model is recommended to reduce the uncertainty

ii. To improve the data quality needed for prediction, an Institutional Warehouse is proposed for purposes of keeping historical snapshots of the transactional database instances which are important for increasing the dimensionality of student data. This is crucial in ensuring the model is fed with dimensional data and not just transactional data that is noisy

iii. Satisfactory index data should also be included in institutional data warehouses detailing not only how the University is ranked but also various ranking indexes on how the students feel about various programs and resources this is to help in enriching the data set required for making enrollment predictions

iv. To ease the use of the proposed model a web application should be adapted to interact with the model for purposes of increasing the model's performance and aiding quick availability of prediction results.

7.3 Suggestions for Future Research

Following the scope and limitations of this research, it was realized that the model does not meet all the institutional needs as far as the admissions process and recruitment decision making is concerned. As such, for future research the following suggestions are made:

i. That the transformation of the data that exist in transactional databases be automated and stored in an institutional warehouse for purposes of consistency of keeping historical snapshots of the databases. At the moment this process is manual and thus might miss out on very crucial changes in instances of data transformation within the transactional databases which makes the data altogether noisy.

ii. That intelligent student scouting should be developed through the coupling of the proposed model and other machine learning algorithms such as clustering that will seek to explain the common features in each classification data set. For instance, a model could be developed that classifies the admitted students as well as explains why a given group of students are likely not to be enrolled by highlighting almost similar shared attributes.

iii. Since the proposed model uses the file-based stub to interact with the users' web application which is hosted on a totally different environment the researcher suggests a better performing protocol of calling procedures remotely other than writing to and reading from stub files so as to increase the performance of the future versions of student enrollment prediction models.

iv. Being that the proposed model only sought to address the low institutional yield issue as well as the uncertainty that arises during the students' enrollment process, it is suggested that future studies delve into exploring prediction in relation to student retention, student progression from one academic year to the next as well as predicting the graduation rates.

v. The proposed model, though of high accuracy could perform better if a series of machine learning algorithms were stacked together to build the model. It is thus suggested that future research could look into using model ensembles on dimensional data to help in improving the prediction accuracy of the model.

References

Abdul Fattah Mashat, M. M. (2012). A Decision Tree Classification Model for University Admission System. *International Journal of Advanced Computer Science and Applications*, 17-21.

Alan Dennis, B. H. (2012). *System Analysis and Design.* Indiana: John Wiley sons Inc.

Albandoz, P. L. (2001). Population and sample. Sampling techniques. *Management Mathematics for European Schools*, 4-16.

Arcilla, R. Q. (2012). Enrollment Forecasting for School Management System. *International Journal of Modeling and Optimization*, 563-566.

Bontrager, B. (2004). Enrollment Management: An Introduction to Concepts and Structures. *College and University Journal*.

Boone, H. N. (2012). Analyzing Likert Data. *Journal of Extension (JOE)*, 7-12.

Borena, M. M. (2014). Higher Education Students' Enrollment Forecasting System Using Data Mining Application in Ethiopia. *HiLCoE Journal of Computer Science and Technology*, 36-43.

C.R, K. (2004). *Research Methodology: Method & Techniques.* New Delhi: New Age International (P) Ltd.

Cabrera, A. F. (1994). Logistic Regression Analysis in Higher Education: An Applied Perspective. *Higher Education: Handbook of Theory and Research*, 225-256.

Chau-Kuang Chen, M. M. (2008). An Intergrated Enrollment Forecast Model. *Association For Institutional Research*, 2-17.

Creswell, J. (2003). *Qualitative, quantitative and mixed methods approaches (2nd ed.).* Thousand Oaks CA: Sage Publishers.

Cristianini, N. a.-T. (2000). *An Introduction to Support Vector Machines: And Other Kernel-Based Learning Methods.* Cambridge,England: England: Cambridge University Press.

Delen, D. (2010). A comparative analysis of machine learning techniques for student retention management. *ELSEVIER*, 498-506.

Diels, H. (2014). *Country Fact Sheet Kenya.* Nairobi: VLIR-UOS.

Furlow, G. (2001). The Case for Building a Data Warehouse. *IEEE.*, 31-34.

Golafshani, N. (2003). *Understanding Reliability and Validity in Research: The Qualitative Report.* Nova SouthEastern University.

Gor, D. R. (2002). *Industrial Statistics And Operational Management.* Delhi.

Gudo, C. (2014). Financing higher education in Kenya: Public – Private partnership approach. *International Journal of Educational Policy Research and Review*, 1-5.

Hanover Research. (2014). *Trends in Higher Education Marketing, Recruitment and Technology.* Hanover Research, Academy Administration Practice.

Harvard University. (2016, September 13). *Harvard Admitted Students Profile*. Retrieved from Admissions Statistics: https://college.harvard.edu/admissions/admissions-statistics

Henriques, S. G. (2015). *The Effect of Shocks to College Revenues on For-Profit.* Washington.

Human Development Resource Center. (2010). *Helpdesk Report: Barriers to Enrollment in Kenya.* Nairobi: UKAID.

Ibrahim Ogachi Oanda, J. J. (2012). University Expansion and the Challenges to Social Development in Kenya:Dilemmas and Pitfalls. *JHEA/RESA*, 49-71.

Inmon, W. (2005). *Building the Data Warehouse.* Indianapolis,Indiana: Wiley Publishers.

Jenkin, J. D. (1982). Prototyping: The New Paradigm for Systems Development. *JSTOR*, 29-44.

Kalekar, P. S. (2004). *Time series Forecasting using Holt-Winters Exponential Smoothing.* Kanwal Rekhi School of Information Technology.

Kalpesh Adhatrao, A. G. (2013). Predicting Students' Performance Using Id3 and C4.5 Classification Algorithms. *International Journal of Data Mining & Knowledge Management Proces*, 39-52.

Ken Peffers, T. T. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, 45-78.

Kenya National Bureau of Statistics. (2015). *Statistical Abstract.* Nairobi: Kenya National Bureau of Statistics.

Kimball Group. (2005). Kimball Dimensional Modelling. *Proceedings of the Eighteenth International Conference on Machine Learning*, (pp. 577-584).

Kumar, R. (2005). *RESEARCH METHODOLOGY: A step-by-step guide for beginners.* London: Sage Publications.

M. D. Orwig, P. K. (1972). *Enrollment Projection Models For Institutional Planning.* Iowa: Act Research Report.

McConell, S. (2003). *Professional Software Development.* Redmond: Microsoft Press.

McIntyre, C. (2007). Performance-Based Enrollment Management. *Annual Forum of the Association for Institutional Research.* Orlando.

Ministry of Education, Science and Technology - Kenya. (2015). *Education For All-2015 National Review.* Nairobi: Ministry of Education Press.

National Research Report. (2012). *Why Did They Enroll? The Factors Infl uencing College Choice.* Chicago: Noel Levitz.

O. Anava, E. H. (2013). Online Learning for Time Series Prediction. *JMLR: Workshop and Conference Proceedings* (pp. 1-13). JMLR.

Ohio University . (2004). *A Comprehensive Analysis of Reasons for First-Year (Freshman) Student Withdrawal at Ohio University.* Ohio.

Owuor, N. A. (2012). Higher Education in Kenya: The Rising Tension between Quantity and Quality in the Post-Massification Period. *Higher Education Studies*, 126-136.

Ozer, P. (2008). *Data Mining Algorithms for Classification.* Nijmegen: Radboud University.

Padmapriya, A. (2012). Prediction of Higher Education Admissibility using Classification Algorithms. *International Journal of Advanced Research in Computer Science and Software Engineering*, 330-336.

Pbnny S. Vissbr, J. A. (1990). *Survey Research Methods.* Belmont, CA: Wadsworth.

Rahman, S. Y. (2012). 'Triangulation' Research Method as the Tool of Social Science Research. *BUP JOURNAL*, 154-163.

Ralph Kimball, M. R. (2002). *The Data Warehouse Tool.* . Newyork: John Wiley & Sons, Inc.

Schutinga, B. J. (2011). *Enrollment management strategies: Effectiveness and usage at member institutions of the Council for Christian Colleges and Universities.* Ames, Iowa: Iowa State University.

Shaweta. (2014). Critical Need of the Data Warehouse for an Educational Institution and Its Challenges. *International Journal of Computer Science and Information Technologies*, 4556-4559.

Strathmore University. (2015, September 13). *University Fact Book.* Nairobi. Retrieved from Institution Data Analysis Unit: https://sagana.strathmore.edu/idau/

Strathmore University. (2016, December 13). *Corporate Facts & Figures.* Retrieved from Strathmore University - Corporate Facts & Figures: http://www.strathmore.edu/en/about-strathmore/corporate-facts-figures

Sulock, M. (2009). *An Application of Binary Logistic Regression to College Admission Data.* Montana: Montana State University.

University of Pennsylvannia. (2013). *Retention and Graduation Rate Analysis.* Pennsylvannia: University of Pennsylvannia Press.

Victoria University. (2013). *Victoria University Student Attrition Rate Analysis.* Melbourne: Victoria University Press.

Vidushi Sharma, S. R. (2012). A Comprehensive Study of Artificial Neural Networks. *International Journal of Advanced Research in Computer Science and Software Engineering*, 278-284.

Vijayalakshmi Sampath, A. F. (2005). *A Logistic Regression Model To Predict Freshmen Enrollments.* Virginia: SAS.

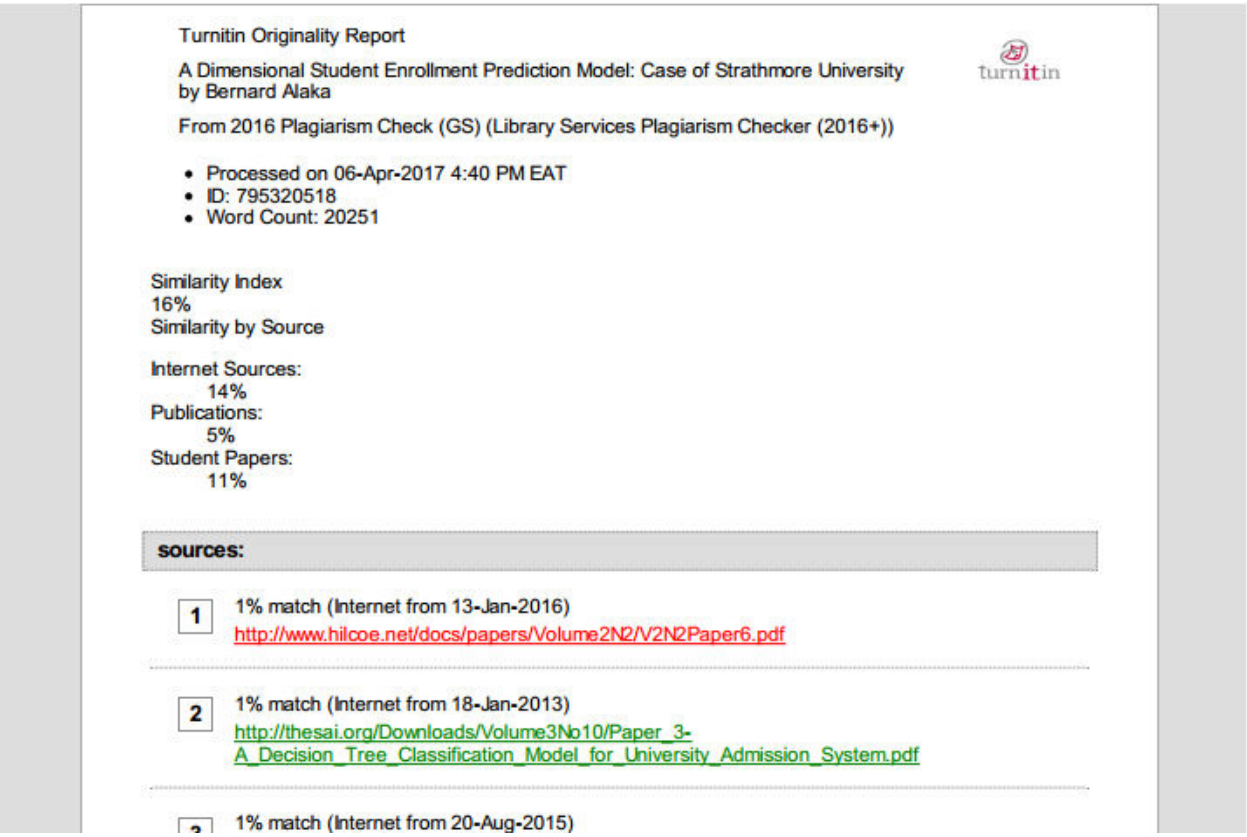
Xie Niuniu, L. Y. (2010). Review of Decision Trees. *IEEE*, 105-109.

Xu, M. I. (2007). . The Effect of Implementation Factors on Data Warehousing Success: An Exploratory Study. *Journal of Information, Information Technology, and Organizations*, 1-11.

Yu, C. T. (2007). Mixed Methods Sampling. *Journal of Mixed Methods Research*, 77-100.

Zaytsev, M. (2011). *Predicting Enrollment Decisions of Students Admited to Claremont McKenna College.* Claremont.

Appendix A: Originality Report



Turnitin Originality Report

A Dimensional Student Enrollment Prediction Model: Case of Strathmore University by Bernard Alaka

From 2016 Plagiarism Check (GS) (Library Services Plagiarism Checker (2016+))

- Processed on 06-Apr-2017 4:40 PM EAT
- ID: 795320518
- Word Count: 20251

Similarity Index
16%
Similarity by Source

Internet Sources:
14%
Publications:
5%
Student Papers:
11%

sources:

1 1% match (Internet from 13-Jan-2016)
http://www.hilcoe.net/docs/papers/Volume2N2/V2N2Paper6.pdf

2 1% match (Internet from 18-Jan-2013)
http://thesai.org/Downloads/Volume3No10/Paper_3-A_Decision_Tree_Classification_Model_for_University_Admission_System.pdf

1% match (Internet from 20-Aug-2015)

Figure A.1: Turn-it-in Originality Report

Appendix B: Interview Guide

i. How fast is the system when the user makes a prediction request or accuracy report?
ii. Does the system classify the admitted students in reasonable proportions based on user experience?
iii. Is there an enrollment prediction model or system that exists within the University. If yes how does it work? If No, what method is used to forecast student enrollment
iv. How likely is the proposed model likely to be adopted in the institutions daily operations in informing decision? Would you accept to use the proposed model?
v. Is the user interface adopted by the system user friendly?
vi. How reliable is the model according to the user in classifying the admitted students

Appendix C: Interview Feedback

i. How fast is the system when the user makes a prediction request or accuracy report?

Fast enough

ii. Does the system classify the admitted students in reasonable proportions based on user experience?

Yes. In most instances

iii. Is there an enrollment prediction model or system that exists within the University. If yes how does it work? If No, what method is used to forecast student enrollment

NO. We rely on experience

iv. How likely is the proposed model likely to be adopted in the institutions daily operations in informing decision? Would you accept to use the proposed model?

Likely.

v. Is the user interface adopted by the system user friendly?

Yes it is. Some addition features in final report e.g. phone number could be included.

vi. How reliable is the model according to the user in classifying the admitted students

60% reliable.

Figure C.1: Interview Sample Feedback

Appendix D: Web Application Screen shot

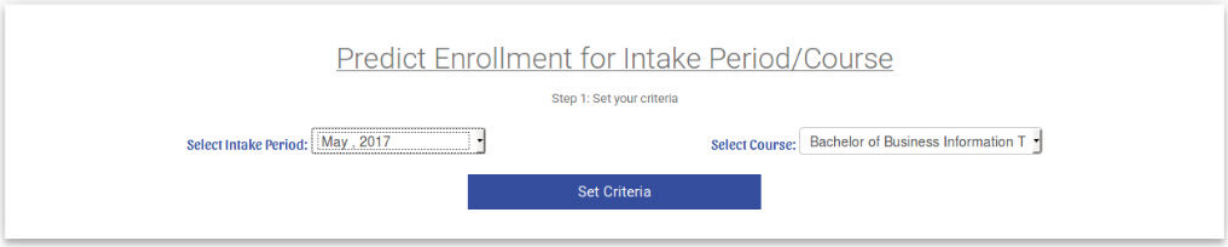


Figure D.1: Web Application step 1 (setting of criteria)

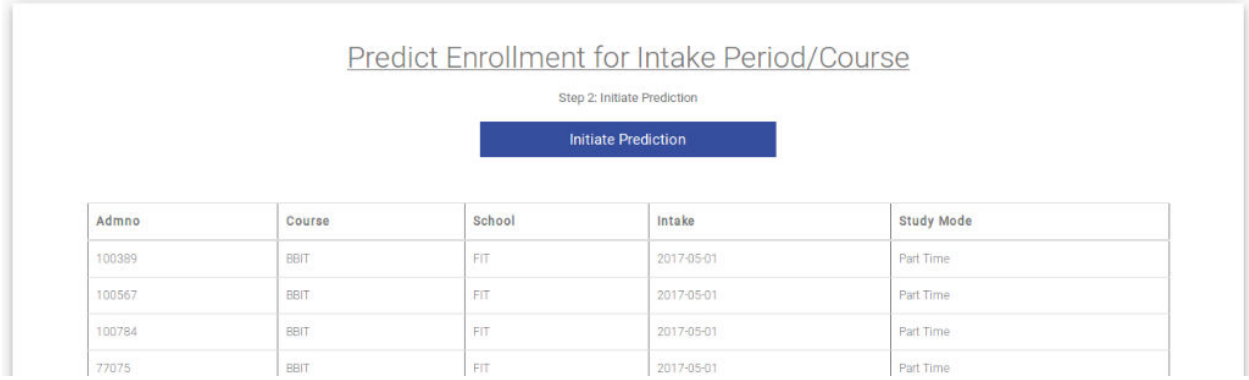


Figure D.2: Web Application step 2 (initiating prediction)

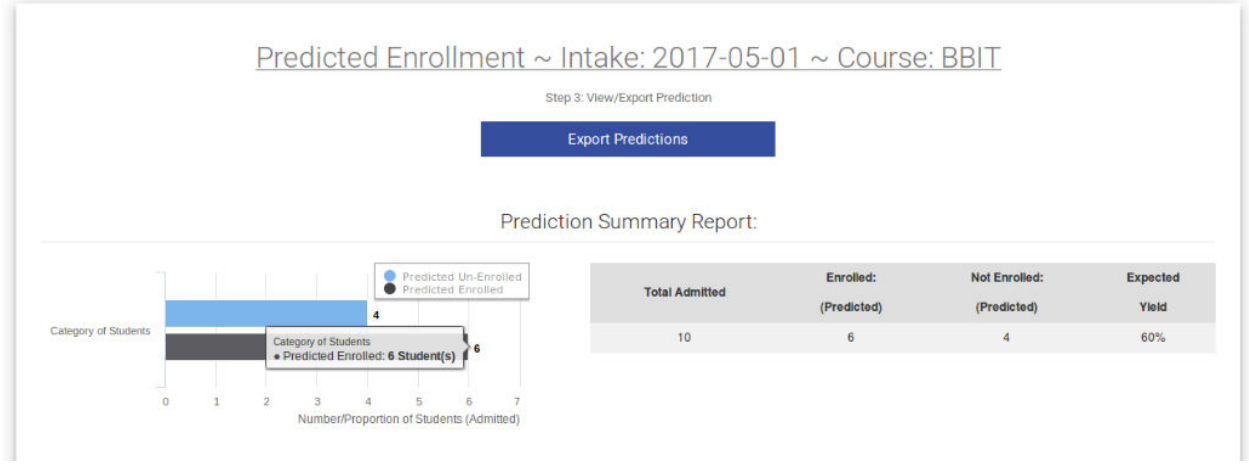


Figure D.3: Web Application step 3 (viewing results and exporting results)

Appendix E: Model Training Code Snippet

```
    optimization." arXiv preprint arXiv:1412.6980 (2014).

    """
    def __init__(self, hidden_layer_sizes=(15,2), activation="tanh",
                 solver='adam', alpha=0.0001,
                 batch_size='auto', learning_rate="constant",
                 learning_rate_init=0.0005, power_t=0.5, max_iter=10000,
                 shuffle=True, random_state=None, tol=1e-4,
                 verbose=False, warm_start=False, momentum=0.9,
                 nesterovs_momentum=True, early_stopping=False,
                 validation_fraction=0.1, beta_1=0.9, beta_2=0.999,
                 epsilon=1e-8):

        sup = super(MLPClassifier, self)
        sup.__init__(hidden_layer_sizes=hidden_layer_sizes,
                     activation=activation, solver=solver, alpha=alpha,
                     batch_size=batch_size, learning_rate=learning_rate,
                     learning_rate_init=learning_rate_init, power_t=power_t,
                     max_iter=max_iter, loss='log_loss', shuffle=shuffle,
                     random_state=random_state, tol=tol, verbose=verbose,
                     warm_start=warm_start, momentum=momentum,
                     nesterovs_momentum=nesterovs_momentum,
                     early_stopping=early_stopping,
                     validation_fraction=validation_fraction,
                     beta_1=beta_1, beta_2=beta_2, epsilon=epsilon)

    def _validate_input(self, X, y, incremental):
        X, y = check_X_y(X, y, accept_sparse=['csr', 'csc', 'coo'],
                         multi_output=True)
        if y.ndim == 2 and y.shape[1] == 1:
            y = column_or_1d(y, warn=True)

        if not incremental:
            self._label_binarizer = LabelBinarizer()
            self._label_binarizer.fit(y)
            self.classes_ = self._label_binarizer.classes_
        else:
            classes = unique_labels(y)
```

Figure E.1: Neural Network training parameters

Appendix F: Use cases and Test cases

Table F.1: Training and Validating Neural Network

Use case: Train Neural Network Model, Validate Neural Network Model	
Primary Actors Administrator	
Preconditions Pre-processed Dimensional student data	
Post conditions A neural network model for student enrollment prediction	
Main Success Scenarios	
Actor Intention	**System Responsibility**
1. Administrator specifies neural network parameters	
2. Administrator specifies number of epochs for model training and validation	
3. Administrator specifies the target class of the dimensional data	
	4. System runs number of specified epochs on dimensional data to train student enrollment prediction model
	5. System successfully generates prediction model and stores it in an executable pickle file
	6. System runs number of specified epochs on dimensional data to validate student enrollment prediction model
	7. System generates validation model and stores it in an executable pickle file

Table F.2: Batch and Individual Student Enrollment Prediction

Use case: Request Batch Student Enrollment Prediction, Request Enrollment Prediction Accuracy Report	
Primary Actors Registrar Administrator	
Preconditions Pre-processed Dimensional student data	
Post conditions An accuracy report for the enrollment prediction model Batch student enrollment prediction report	
Main Success Scenarios	
Actor Intention	**System Responsibility**
1. Administrator or Registrar specifies intake period for batch students	
2. Administrator or Registrar specifies the course(s) for batch students	
	3. System runs student enrollment prediction for specified students given the intake period and course
	4. System generates prediction model accuracy
Extensions	
At any point the system fails to generate batch student prediction Re-enter a valid intake for the desired batch students	

Table F.3: Individual and Batch Enrollment Prediction use case

<table>
<tr><td colspan="2">Use case: Request Individual Student Enrollment Prediction, Request Batch Student Enrollment Prediction</td></tr>
<tr><td colspan="2">Primary Actors
Admissions Staff</td></tr>
<tr><td colspan="2">Preconditions
Pre-processed Dimensional student data</td></tr>
<tr><td colspan="2">Post conditions
Batch student enrollment prediction report
Individual student enrollment prediction report</td></tr>
<tr><td colspan="2">Main Success Scenarios</td></tr>
<tr><td>Actor Intention</td><td>System Responsibility</td></tr>
<tr><td>1. Admissions Staff specifies intake period for batch students</td><td></td></tr>
<tr><td>2. Admissions Staff specifies the course(s) for batch students</td><td></td></tr>
<tr><td>3. Admissions Staff specifies the applicant number for specific individual student</td><td></td></tr>
<tr><td></td><td>4. System runs batch student enrollment prediction for specified students given the intake period and course</td></tr>
<tr><td></td><td>5. System runs individual student enrollment prediction for specified students given student’s applicant number</td></tr>
<tr><td colspan="2">Extensions</td></tr>
<tr><td colspan="2">At any point the system fails to generate batch or individual student prediction
Re-enter a valid intake for the desired batch students or re-enter a valid student applicant number</td></tr>
</table>

Table F.4: System Testing

Test Case	Check Criteria	Priority
Performance	How long does it from the moment a user request for a prediction or accuracy report from the system	High
Functionality	Does the system correctly write and update the criteria.dat stub file for purposes of consistency with querying the warehouse	High
Integration	Does the system pass parameters correctly to the model and pull them in the right format	High

Table F.5: Model and System Test Results

Test Case	Test Results	Remarks
Functionality	PASS	Model predicts student enrollment to the correct accuracy and System updates stub files appropriately
Integration	PASS	The Model and the System work together seamlessly
Load Testing	PASS	The Model while training reaches convergence in a short time
Performance	PASS	The system takes a short time to display prediction results
Reliability	PASS	The accuracy of the model is consistently high

Table F.6: User Acceptance Testing

Test Case	Check Criteria	Priority
Performance	How fast is the system when the user makes a prediction request or accuracy report	High
Functionality	Does the system classify the admitted students in reasonable proportions based on user experience	High
Acceptability	Is there an enrollment prediction model or system that exists within the University. If Yes how does it work? If No, what method is used to forecast student enrollment	High
Acceptability	How likely is the proposed model likely to be adopted in the institutions daily operations in informing decision? Would you accept to use the proposed model?	High
Aesthetic Value	Is the user interface adopted by the system user friendly	High
Reliability	How reliable is the model according to the user in classifying the admitted students	High