

# CHAPTER ONE

## INTRODUCTION

### 1.1 Background of the Study

Academic performance or "academic achievement" is the extent to which a student, teacher or institution has attained their short or long-term educational goals. Commonly measured through examination or continuous assessment, there is however no general agreement on how it is best evaluated or which aspects are most important. Academic achievement plays an important role in determining the worth of graduates who will be responsible for the social and economic growth of the country.

Educational Data Mining (EDM) is a new discipline, focusing on studying the methods and creating models to utilize educational data, using those methods to better understand students and their performance. Educational data mining has become a vital need for academic institutions to improve the quality of education. Kumar et al., 2011 analyzed it as the process of transforming raw data compiled by educational systems to useful information that can be used to take informed decisions and answer research questions. Educational data mining methods have been successful at modeling a range of research relevant to student learning in online intelligent systems. Models also achieve better accuracy every year and are being validated to be more generalizable over time. Research in education has resulted in several new pedagogical improvements. Computer-based technologies have transformed the way we live and learn. Today, the use of data collected through these technologies is supporting a second-round of transformation in all areas and learning with different achievements.

Baker and Yacef (2009) summarized the four goals of educational data mining:

- Predicting students' future learning behavior by creating student models that categorizes a students characteristics or states that make up the students' knowledge, motivation, meta-cognition and attitudes
- Discovering or improving knowledge domain models that explains the interrelationship between a knowledge in a domain and the materials that characterize the content to be learned

- Studying the most effective pedagogical support for students learning that can be achieved through learning systems.
- Establishing empirical evidences to support pedagogical theories, framework and educational phenonena to determine core influential components of learning to enable the designing of better learning systems.

Educational Data Mining involves the application of data mining techniques to the following educational problems

1. Providing Feedback for Supporting Instructors
2. Student modeling,
3. Detecting Student Behavior
4. Predicting Student's Performance
5. Recommendations for Students
6. Grouping students
7. Constructing Courseware
8. Planning and Scheduling
9. Students Social Network Analysis

Predictive modeling is the general concept of building a model that is capable of making predictions. Typically, such a model includes a machine learning algorithm that learns certain properties from a training dataset in order to make those predictions. Predictive modeling can be divided further into two sub areas: Regression and pattern classification. Regression models are based on the analysis of relationships between variables and trends in order to make predictions about continuous variables, e.g., the prediction of the maximum temperature for the upcoming days in weather forecasting. In contrast to regression models, the task of pattern classification is to assign discrete class labels to particular observations as outcomes of a prediction.

Predictive modeling requires four components; the methodology followed to deploy the model, the data mining techniques adopted to build the model, input attributes used by the model and the performance metrics used to evaluate the system.

Academic performance prediction involves analysis and involvement of educational data mining techniques for the purpose of predicting student's performance. Based on prediction results, if the student needs are fulfilled timely, then the overall result and performance will increase year by year. The success of any educational institute depends upon the success of the students of the Institution. Student's performance prediction and its analysis are essential for improvement in various attributes of students like final grades, attendance etc. This prediction helps teachers in identification of weak students and to help them improve in the studies. Improvement of student performance and enhancement of quality of education is of utmost importance for all educational institutions.

For the purpose of performance analysis and prediction, important attributes and previous records of students are gathered. Subsequently, various data mining techniques are applied to get deeper insights and predictions. Data mining techniques refers to the algorithms used in extracting data from a large repository of data. In recent years, various data mining techniques have been used such as Naïve Bayes, Decision tree, Nearest neighbor, support vector machine, neural networks, outlier's detections and advanced statistical techniques. These techniques are applied on the student data in order to get information, to help in decision support systems, and pattern extracting etc.

Universities focus on the most important information in the data they have collected about the behavior of their students and potential learners. Data mining involves the use of data analysis tools to discover previously unknown, patterns and relationships in large data sets. These tools can include statistical models, mathematical algorithms and machine learning methods. These techniques are able to discover information within the data that queries and reports can't effectively reveal.

Academic advising is a decision-making process by which students realize their maximum educational potential through communication and information exchanges with an advisor. The purpose of academic advising is to assist students in the development of meaningful educational and career goals. Academic advisors assist students in developing educational plans that will help them achieve their life goals. Academic advisors at the university level provide information about academic progress and degree requirements, and carefully review students' academic and educational needs, performance, and challenges.

The differential students' performance in tertiary institutions is a source of great concern and research interest to the higher education managements, government, parents and other stakeholders because of the importance of education to national development. There is need to extract useful information from the available students' large dataset and inform academic policies on how best to improve student retention rates, allocate teaching and support resources, or create intervention strategies to mitigate factors that affect student performance (Kuyoro et al., 2013). Maximizing the potential of students, providing evidence of delivering value for money to the bodies that fund them, and performing up to expectation is very crucial to tertiary institutions. Most institutions are often judged by the quality of the awards they provide; for instance, the more honours level graduates a course provides, the better the course is perceived to be. This provides additional quest for institutions to take proactive steps to investigate students' data with a view of finding useful information that can aid planning activities, decision making and students' intervention strategies. It is necessary to carefully measure student outcomes or expected outcomes that may provide evidence as to whether student potential is being realized against some benchmarks (Kuyoro et al., 2013)

Student's academic performances are affected by many factors, like personal, socio-economic and other environmental variable (Baradwaj 2011). Knowledge about these factors and their effect on student performance can help in managing their effect. According to Ventura and Romero, (2011), poor performance of students in tertiary institutions has been partly traced to poor academic background and wide range of other predictors, including personality factors, intelligence, gender, academic achievement, previous college achievements, and demographic data. Many researchers have come to some interesting conclusion as to which of these predictors has impacted students' academic performance in tertiary institutions. There is a growing interest and concern in many countries about the problem of school failure and the determination of its main contributing factors. This problem has been referred to as "the one hundred factors problem". Different predictors including gender, personality factors, intelligence, aptitude tests, academic achievement, previous college achievements, and demographic data have been identified in literature as contributors to students' academic performance.

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn, identify patterns, make decisions and improve from experience without being explicitly programmed. The primary aim of machine learning is to develop computer

programs that can access data and use it learn for themselves without human intervention or assistance. Machine learning is divided into supervised, unsupervised, semi supervised and reinforcement learning

Supervised learning is the machine learning task of inferring a function from labeled training data. Labeled data is a dataset that contains both the input and the output data. The training data consist of a set of training examples. In supervised learning, each example is a pair consisting of an input object (typically a vector) and a desired output value (also called the supervisory signal). A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples. Unsupervised learning is the training of machine using information that is neither classified nor labeled and allowing the algorithm to act on that information without guidance. Here the task of machine is to group unsorted information according to similarities, patterns and differences without any prior training of data. Semi-supervised learning is a class of machine learning tasks and techniques that also make use of unlabeled data for training – typically a small amount of labeled data with a large amount of unlabeled data. Semi-supervised learning falls between unsupervised learning (without any labeled training data) and supervised learning (with completely labeled training data). Many machine-learning researchers have found that unlabeled data, when used in conjunction with a small amount of labeled data, can produce considerable improvement in learning accuracy. Reinforcement learning is a kind of machine learning in which artificial intelligent agents attempt to find the optimal way to accomplish a particular goal, or improve performance on a specific task. As the agent takes action that goes toward the goal, it receives a reward. The overall aim: predict the best next step to take to earn the biggest final reward.

Classification is a supervised learning approach in which the computer program learns from the data input given to it and then uses this learning to classify new observation. This data set may simply be binary class (like identifying whether the person is male or female or that the mail is spam or non-spam) or it may be multi-class too (identifying if an element is high, low or medium).

Regression is a supervised learning approach used to predict a continuous value. A regression problem is when the output variable is a real value, such as “dollars” or “weight”. Predicting

prices of a house given the features of house like size, price etc is also one of the common examples of regression.

Hybrid machine learning systems combine or integrate different machine learning models. Since each machine learning method works differently and exploits a different part of problem (input) space, usually by using a different set of features, their combination or integration usually gives better performance than using each individual machine learning or decision-making model alone. Hybrid models can reduce individual limitations of basic models and can exploit their different generalization mechanisms.

Intelligent agents are autonomous entities which act upon an environment using sensors and acts upon it through actuators or effectors (Woolridge, 2002). A human agent has eyes, ears, and other organs which work for sensors and hand, legs, vocal tract work for actuators. A robotic agent can have cameras, infrared range finder, NLP for sensors and various motors for actuators. Agents are task-oriented, active, modeled to perform specific tasks and capable of autonomous action and decision-making. When combining multiple agents in one system to solve a problem, the system becomes a Multi-Agent System (MAS). These systems are comprised of agents that solve problems individually that are simpler than the overall problem. They can communicate and assist each other in achieving larger and more complex goals. Agents and data mining can work together to achieve required target.

Data mining and intelligent agents have emerged as two fields with immense potential for research. Every intelligent agent is self-sufficient, acting independently within its boundary while collaborating with other agents to perform the assigned task efficiently. The ability of agents to learn from their experience complements the data mining process. Agent mining helps to overcome the challenges faced by data mining in a distributed heterogeneous environment. Data mining agents perform various functions of data mining. It is increasingly significant to develop better methods and techniques to organize the data for better decision-making processes (Albashiri, 2010). The distributed nature of agent mining brings several advantages to data mining such as autonomy, scalability, reliability, security, interactivity and high speed (Fariz et al., 2015). Agents can be used to automate the various tasks like data selection, data cleansing, and data pre-processing, to perform classification, clustering and knowledge representation. As an emerging area, a lot of research can be performed in this field.

A data mining agent is a pseudo-intelligent computer program designed to find out specific types of data, along with identifying patterns among those data types. These agents are typically used to detect trends in data, alerting organizations to paradigm shifts so effective strategies can be implemented to either take advantage of or minimize the damage from alterations in trends. In addition to reading patterns, data mining agents can also "pull" or "retrieve" relevant data from databases, alerting end-users to the presence of selected information.

In the last few years, agent technology has come to the forefront in the software industry because of the advantages that Multi-agent systems have in complex and distributed environments. Multi-agent systems (MAS) are commonly intended as computational systems where several autonomous entities called agents, interact or work together to perform some tasks. In MAS, communication enables the agents to exchange information on the basis of which they coordinate their actions or cooperate with each other and this is done through Agent Communication Languages (ACL). (Yasser et al., 2015)

Designing a predictive model requires a data set that has the essential attributes to predict future academic performance. After careful consultation with relevant experts, questionnaire that contains most relevant attributes regarding academic attainments of the students was designed. The survey consists of 38 questions amongst which spanned demographic/socio economic questions, academic related question, work related questions and social related questions.

## **1.2 Statement of the Problem**

Provision of quality education to students and to improve the quality of managerial decisions is the major objective in any academic institution. Information is requested from students from time to time and the institutions data bank is updated periodically with such information. These data bank are however hardly used to improve decision making. The discovered knowledge can be used to offer a helpful and constructive recommendations to the academic planners in higher education institutes to enhance their decision making process, to improve students' academic performance and trim down failure rate, to better understand students' behavior, to assist instructors, to improve teaching and many other .

Some of the problems highlighted include:

- Difficulty in selecting the best machine learning classification model for classifying student's academic performance with a significant accuracy rate.
- Identifying the main key indicators that could help in creating the classification model for predicting students' dissertation project grades
- Selecting the right variables/attributes for correct prediction
- Using the right predictive technique and tools to discover hidden characteristics for early identification of "at risk" students and help them.

The problem of accurate student performance prediction is still a challenging task due to various issues and many other factors are involved in it. This work thereby addresses the capabilities of educational data mining in improving the quality of decision making process in higher learning institution by proposing the model of student performance predictor.

### **1.3 Aim and Objectives of the Study**

The aim of this study is to develop a predictive model for classifying students' academic performance.

The objectives that are considered here are to:

- Identify and analyze the different factors that assumed to affect students performance, identifying those which have the biggest impact on their academic performance.
- Provide a platform for classification of students' performance into High, Medium or Low category and offer academic advising based on their performance.
- Provide an effective database that can query student's academic standing and their personal attributes.
- Increase the efficiency of the prediction system and reduce execution time.

### **1.4 Significance of the Study**

The system offers enormous benefits to the following users:

1. Lecturers/ Academic Advisors: The prediction model will help teachers and tutors identify weak and strong students so teachers can lay more emphasis on instructions and procedures when dealing with the weak students so as to enhance overall academic



performance. An academic advisor can refer to the prediction results when giving advice to students who perform poorly in their studies so that preventive measures can be taken much earlier.

2. Department and Faculty: Curriculum committees can use prediction results to guide changes to the curriculum and evaluate the effects of those changes.. In addition, an instructor can further improve his/her teaching and learning approach, as well as plan interventions and support services for weak students.
3. University: Academic Performance is an important factor people consider before applying for Postgraduate Studies. An institution that is known for producing low performance postgraduates is at risk of having low intakes. The need for Prediction Performance System comes up as this will help in the early prediction of weak students and help them to focus on their weak areas. The result from academic performance prediction can also be used to formulate policy that students who have no tendency of doing well in school be discovered at early stage of academic pursuit, thereby preventing continuous waste of human and material resources on such non-productive students or suggesting Departments that they could fit into
4. Parents/Guardian/partners: Results have shown that Parents/Guardian and Partners have effects on the academic performance of Students. The Study helps to analyse the influence of family background on student's performance predictions. Attributes such as size of family, encouragement/motivation, from parents/spouses/siblings, highest qualification of sponsor and other factors will help determine those factors that affect performance. This will help in proffering solutions to those problems
5. Government: Recent multi-country case studies have highlighted the capacity of quality assurance in higher education to support nation-building in multiple ways, ranging from promoting a more open and transparent society to supporting economic goals and increasing graduate employability. Education is vital for economic development.

## **1.5 Scope of the Research**

The scope of this research is to create a students' performance prediction model by using psychometric factors of students as variable predictors and hybridizing Naïve Bayes and K-Nearest Neighbour as Classifiers. The sample data of this research came from student academic databases and the surveyed intrinsic motivation and behavior of Postgraduate students of Accountancy Department, Nnamdi Azikiwe University Awka. The scope of the research is limited to the investigation of the effects of a student's prior achievement, domain-specific prior knowledge, and learning progression on their academic performance in the Masters course. Demographic factors, academic and Work Related Factors and Social were included in constructing the predictive model.

## **1.6 Limitations of the Study**

Although the research has achieved its aims, there were some unavoidable limitations that should be discussed

- a. The work deals with supervised learning. This means that you have to manually calculate the class for all the data used in the training point, thereby taking a lot of computational time.
- b. Extra care had to be taken in calculating the training data because if we give an input which is not from any of the classes in the training data, then the output may be a wrong class label.
- c. In assigning the classes using Naive Bayes algorithm, care had to be taken because if the category of any categorical value is not seen in the training data set, the model assigns a zero probability to the category and then prediction cannot be made.
- d. Lack of easy access to reputable journals – the researcher's limited access to certain journals considered reputable and from trusted reliable online libraries was a serious impediment to this study. Journals like those of IEEE and Elsevier, in the IEEE and Science Direct digital libraries respectively were not possible to access except with full subscription and payment. Some of the papers belonging notable libraries which were accessed had only their abstract available and provided vague information about their content.
- e. Reliability and Validity of the information filled by the students in the questionnaire could not be ascertained.

### 1.6.1 Definition of Terms

**Agents:** Agents are sophisticated computer programs that act autonomously on behalf of their users, across open and distributed environment to solve a growing number of complex problems.

**Agent Communication Language (ACL):** ACL is a language that provides a set of application-independent primitives that allow an agent to state its intention in an attempt to communicate with other agents.

**Data Mining:** The process of discovering patterns in large data sets involving methods at the intersection of machine learning, statistics, and database systems

**Educational Data Mining:** An emerging discipline concerned with developing methods for exploring the unique types of data that come from educational settings and using those methods to better understand students, and the settings which they learn in

**Artificial Intelligence (AI):** Intelligence exhibited by any manufactured system

**Distributed Artificial Intelligence (DAI):** Distributed Artificial Intelligence is a subfield of AI research dedicated to the development of solutions for complex problems that are not easily solvable with classic algorithmic programs. There are three main streams in DAI research: Parallel program solving, Distributed program solving, and Agent-based problem solving

**Intelligent Agents:** is a program that gathers information or performs some other service without your immediate presence and on some regular schedule.

**Machine Learning** are computer programs that can learn from experience with respect to some class of tasks and performance measure.

**Multi-Agent Systems (MAS):**Multi-Agent Systems (MAS) are systems composed of multiple agents

**Model:** An abstract and simplified representation of a given reality, either already existing or just planned

## **CHAPTER TWO**

### **LITERATURE REVIEW**

#### **2.1 Theoretical Framework**

Measuring of Students' academic performance hinges on diverse factors like personal, socio-economic, psychological and other environmental variables. Commonly students academic performance is measured by previous Cumulative Grade Point Average (CGPA) but there are various other important attributes that affect the overall performance of the result. Prediction models that include all these variables are necessary for the effective prediction of the performance of the students. The prediction of student performance with high accuracy is beneficial to identify the students with low academic achievements. The identified students can be individually assisted by the educators so that their performance can improve in future.

##### **2.1.1 Data Mining**

Wu et al., (2008) described Data Mining (DM) as a powerful artificial intelligence (AI) tool, which can discover useful information by analysing data from many angles or dimensions, categorize and summarize the relationships between and is then used to make improved decision. In Data Mining solutions, algorithms can be used either independently or together to achieve the desired results. Some algorithms can explore data; others extract a specific outcome based on that data. For example, clustering algorithms, which recognize patterns, can group data into different n-groups. The data in each group are more or less consistent, and the results can help create a better decision model. Multiple algorithms, when applied to one solution, can perform separate tasks. For example, by using a regression tree method, they can obtain financial forecasts or association rules to perform a market analysis.

A large amount of data in databases today exceeds the human ability to analyse and extract the most useful information without help from automated analysis techniques. Data mining has discovered patterns with respect to a user's needs. The accurate discovery of patterns through data mining is influenced by several factors, such as sample size, data integrity, and support from domain knowledge, all of which affect the degree of certainty needed to identify patterns. Typically, data mining uncovers a number of patterns in a database; however, only some of them are interesting. Useful knowledge constitutes the patterns of interest to the user. It is important for users to consider the degree of confidence in a given pattern when evaluating its validity.

Knowledge Discovery in Database (KDD) is a process, which is used to extract useful information by performing various actions on the dataset. Data mining and KDD are often used interchangeably to serve the purpose of mining. Data Mining is one of the steps involved in the KDD process while the KDD is the overall process of mining. (Garima and Santosh., 2017). The steps involved in Knowledge discovery in Database include data processing which the important features are selected, normalization in which involves transforming all the variables in the same range (categorizing the data), data subletting which involves choosing the right attributes from a whole range of attributes. Data mining involves extraction applying machine learning techniques such as classification, regression etc on the data to discover hidden patterns. Post processing involves visualizing the knowledge, interpretation of the result as shown in figure 2.1.

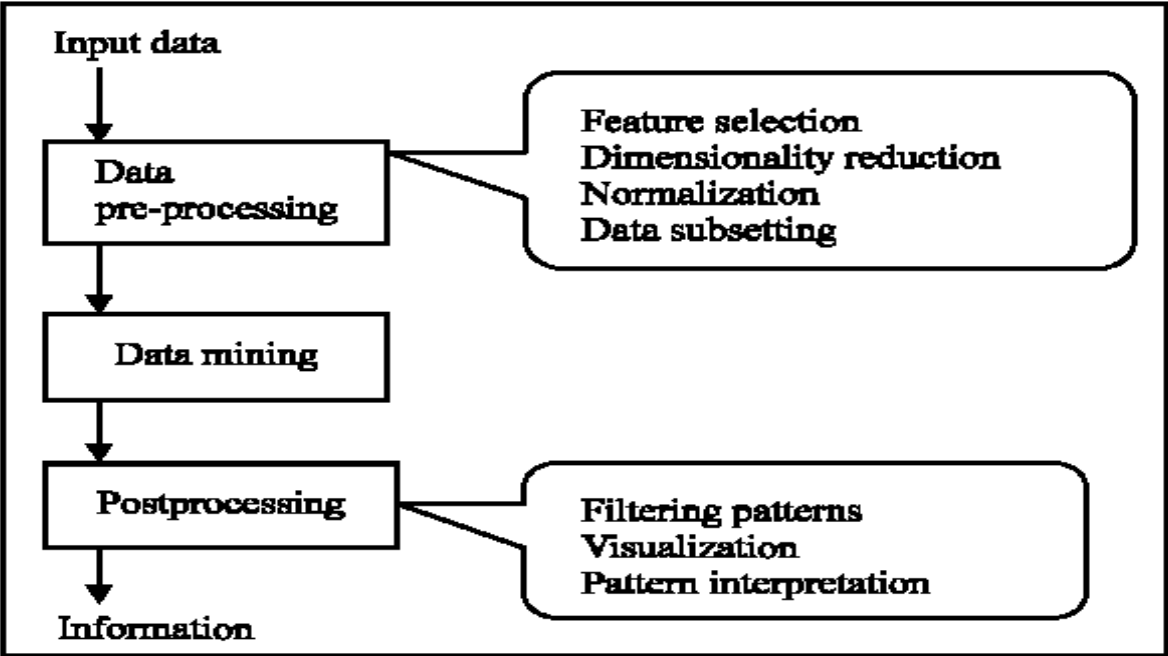


Figure 2.1: The Process of Knowledge Discovery in Database (Paris et al.,2010)

**2.1.1.1 Data Mining Process**

It is important to realize that the problem of discovering or estimating dependencies from data or discovering new data is only one part of the general experimental procedure used by engineers, scientists and others who apply standard steps to draw conclusions from data (Kantardzic 2003).

The overall process of finding and interpreting patterns and models from data involves the repeated application of the following steps.

1. Data Collection: Determining how to find and extract the right data for modeling. First, we need to identify the different data sources that are available. Data may be scattered in different data “silos,” spreadsheets, files, and hard-copy (paper) lists (Nisbet et al., 2009).
2. Data integration: Integration of multiple data cubes, databases or files. A big part of the integration activity is to build a data map, which expresses how each data element in each data set must be prepared to express it in a common format and record structure (Nisbet et al., 2009).
3. Data selection: First of all the data are collected and integrated from all the various sources, and we select only the data which are useful for data mining. Only relevant information is selected.
4. Pre-processing: The Major Tasks in Data Pre-processing are: Cleaning, Transformation and Reduction.
  - i. Data cleaning: Also called data cleansing. It deals with errors detecting and removing error from data in order to improve the quality of data. Data cleaning usually includes fill in missing values and identifying or removing outliers.
  - ii. Data Transformation: Data transformation operations are additional procedures of data pre-processing that would contribute toward the success of the mining process and improve data-mining results. Some of Data transformation techniques are Normalization, Differences and ratios and Smoothing (Kantardzic., 2003).
  - iii. Data Reduction: For large data sets, there is an increased likelihood that an intermediate, data reduction step should be performed prior to applying data-mining techniques. While large datasets have potential for better mining results, there is no guarantee that they will produce better knowledge than small datasets. Data reduction obtains a reduced dataset representation that is much smaller in volume, yet produces the same analytical results.
5. Building the model: In this step appropriate data mining task (example association rules, sequential pattern discovery, classification, regression, clustering, etc.), the data mining technique and the data mining algorithm(s) are chosen and implemented to build the model.

6. Interpretation of the discovered knowledge (model /patterns): The interpretation of the detected pattern or model reveals whether or not the patterns are interesting. This step is also called Model Validation/Verification and uses it to represent the result in a suitable way so it can be examined thoroughly (Kantardzic 2003).
7. Decisions / Use of Discovered Knowledge: It helps to make use of the knowledge gained to take better decisions. A schematic diagram of the data mining process is illustrated in figure 2.2.

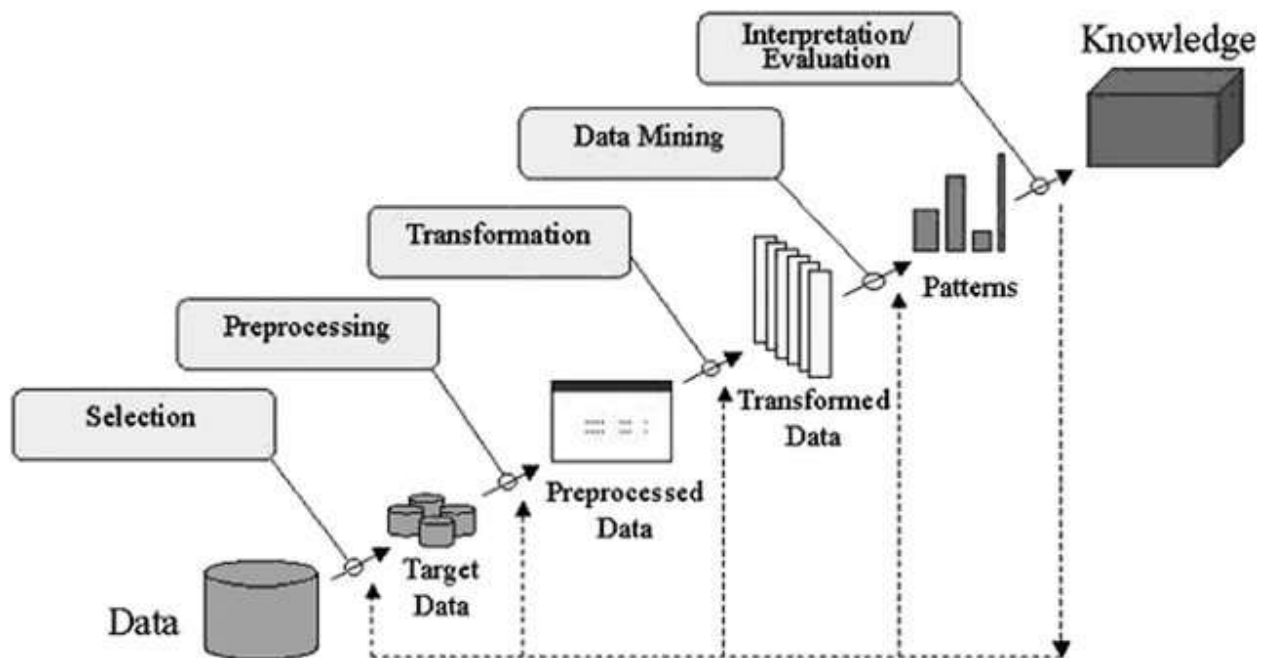


Figure2.2: Data Mining Process (Fayyad et al., 1996)

### Data Mining Tasks

Tasks well suited to data mining include the following:

1. Prediction- determining the value of one variable based on patterns found in others.
2. Classification- dividing the data into predefined categories based on their attributes.
3. Clustering- finding similarities and differences in a data set's attributes in order to identify a set of cluster to describe the data. The cluster may be mutually exclusive and exhaustive or consist of overlapping categories.
4. Description- putting a given data pattern or relationship into human interpretable form (Fayyad et al., 1996).

### **2.1.1.2 Educational Data**

Decision-making in the field of academic planning involves extensive analysis of huge volumes of educational data. Data are generated from heterogeneous sources like diverse and distributed, structured and unstructured data. These data are mostly generated from the offline or online sources:

**i. Offline Data.**

Offline Data are generated from traditional and modern classroom, Interactive teaching/learning environments, learner/educators information, students attendance, Emotional data, Course information, data collected from the academic section of an institution etc..

**ii. Online Data.** (Jindal and Borah., 2013)

Online Data are generated from the geographically separated stake holder of the education, distance educations, web based education and computer supported collaborative learning used in social networking sites and online group forum. E.g.: Web logs, E-mail, Spread sheets, and Tran scripted Telephonic Conversations, Medical records, Legal Information, Corporate contracts, Text data, publication databases etc

### **2.1.1.3 Educational Data Mining**

Educational data mining (EDM) is describes a research field with the application of data mining, machine learning and statistics to information generated from educational settings e.g., universities and intelligent tutoring systems. EDM is an emerging discipline, concerned with developing methods for exploring the unique types of data that come from educational settings, and using those methods to better understand students, and the settings which they learn in (Algarni., 2016).

There are two fields that have been identified as focused on analyzing educational data with a view to understand learners and their learning environment. These fields are: Educational data mining, and Learning analytics. (Siemens and Baker, 2012)

Learning Analytics is the measurement, collection, analysis and reporting of data about learners and their contexts for the purpose of understanding and optimizing learning and the environments in which it occurs.



Although the two fields have similar focus, to understand the students and the environment in which they learn, EDM is usually preferred for the following reasons:

EDM focuses more on automated discoveries. EDM is preferred since the knowledge discovery is purely data driven after the humans have contributed the initial set of features (Mgala and Mbogho, 2015).

Secondly, EDM follows a preferred framework where features are studied together but also in smaller groups called feature subsets. This way, it is possible to also determine the features that have the greatest impact on the target class (Bratu et al., 2008). This could also be beneficial to education stakeholders who would like to come up with strategic measures; focusing on a subset of features can be more meaningful.

Further, EDM is preferred because of its popularity with the community that have conducted student academic performance prediction (Bhardwaj and Pal, 2012, Golding and Donaldson, 2006, Kotsiantis et al., 2004).

Lastly, EDM is more focused on automation that could empower more educational stakeholders; it has a bias towards automated adaptation. This is in line with the goal of the study, where a system that is built will be used by the education stakeholders to predict the future performance of new students. That way, the automation will allow for the technology to be available and usable by a wider population of the education stakeholders. Additionally, EDM uses approaches and methods relevant to students predictive model, such as binary classification techniques that groups students into the desired categories (Bhardwaj and Pal, 2012).

Educational Data Mining has been looked at as a process of applying Data Mining techniques to data originating from the education sector with a view to resolving educational issues (Romero and Ventura, 2010). EDM uses Data mining as a means to detect useful and meaningful patterns from data (Romero and Ventura, 2010). The aspect of being an emerging field comes about because the DM techniques are lately being used in the area of education.

#### 2.1.1.4 Goals of Educational Data Mining

Educational Data Mining aims to improve several aspects of educational system. EDM Objectives depend on the view-point of the final users (learner, educator, administrator and researcher). Goals of Educational Data Mining include:

1. **Student Modeling:** User modeling in the educational domain incorporates such detailed information as student's characteristics or states such as knowledge, skills, motivation, satisfaction, meta-cognition, attitudes, experiences and learning progress, or certain types of problems that negatively impact their learning outcomes. The common objective of students modeling is to create or improve a student model from usage information.
2. **Predictive Modeling:** This involves predicting students' performance and learning outcomes. The objective is to predict a student's final grades or other types of learning outcomes (such as retention in a degree program or future ability to learn) based on data from course activities. Building a predictive model requires four important components.
  - The Methodology followed to deploy the model.
  - Techniques adopted to build the prediction model.
  - Input attributes used by the model and
  - Performance Metrics used to evaluate the model.
3. **Generating Recommendations:** The objective is to recommend to students the content (or tasks or links) that is most appropriate for them at the current time.
4. **Analysing learner's behaviour:** This involves applying educational data mining techniques to analyse learner behaviour.
5. **Maintaining and improving courses:** The objective is to determine how to improve courses (contents, activities, links, etc.) by using information about student usage and learning. It also involves discovering or improving models that characterize the subject matter to be learned (e.g. math, science, etc.), identifying fruitful pedagogical sequences, and suggesting how these sequences might be adapted to student's needs.
6. **Learners:** This includes supporting a learner's reflections on the situation, providing adaptive feedback or recommendations to learners, responding to student's needs and improving learning performance.

7. Educators: To understand their student's learning processes and reflect on their own teaching methods, to improve teaching performance, to understand social, cognitive and behavioural aspects, etc.
8. Administrators: To evaluate the best way to organize institutional resources (human and material) and their educational system (Jindal and Borah., 2013)

### 2.1.1.5 Educational Data Mining Process Phases

1. The first phase of educational data mining is to find the relationships between the data of educational environment using data mining techniques i.e. classification, clustering, regression etc.
2. The second phase of educational data mining is validation of discovered relationships between data so that uncertainty can be avoided.
3. The third phase is to make predictions for future on the basis of validated relationships in learning environment.
4. The fourth phase is supporting decision making process with the help of predictions (Upadhyay and Katiyar., 2014)

### 2.1.1.6 Educational Data Mining Techniques

Educational Data Mining not only applies to data mining techniques such as classification, clustering, and association analysis, but also applies to the methods and techniques drawn from the variety of areas related to Educational Data Mining (statistics, machine learning, text mining, web log analysis, etc.). The data mining methods are illustrated in figure 2.3.

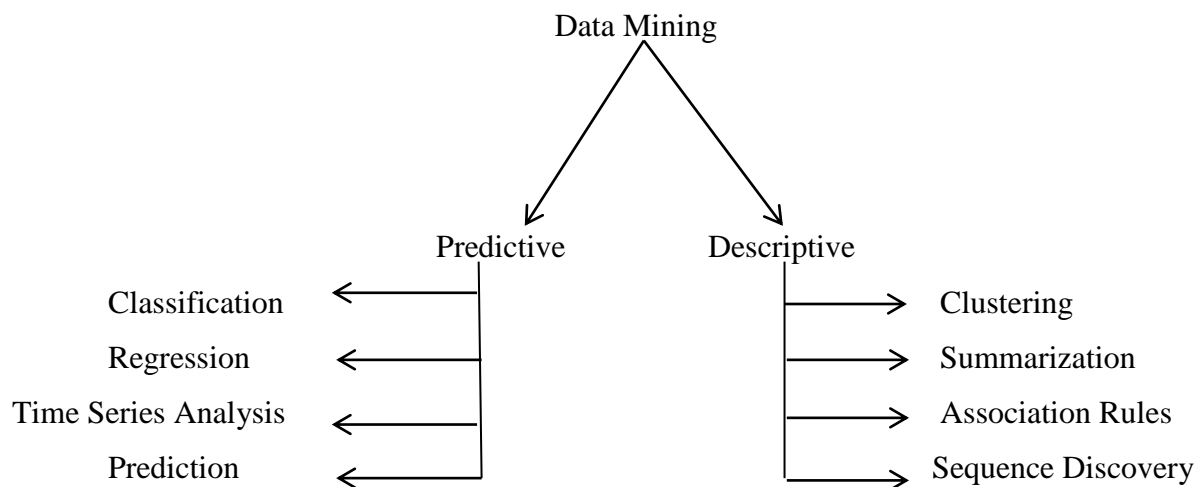


Figure 2.3: Data Mining Methods (<https://www.wideskills.com/data-mining-tutorial>)

A. Predictive Data Mining methods: This method is used for the analysis of pattern or information in the data set which are unknown and also used in the prediction of future values to know what would likely happen in future by analysing the current dataset. Predictive Data Mining involves several techniques described as follows:

- a) Prediction: Prediction task predicts the possible values of missing or future data. Prediction involves developing a model based on the available data and this model is used in predicting future values of a new data set of interest. For example, a model can predict the income of an employee based on education, experience and other demographic factors like place of stay, gender etc. Also prediction analysis is used in different areas including medical diagnosis, fraud detection etc.
- b) Regression: Regression is an inherently statistical technique used regularly in data mining. Regression analysis establishes a relationship between a dependent or outcome variable and a set of predictors. Regression is a supervised learning data mining technique in which the database is partitioned into training and validation data. There are two type of regression technique: Linear and Non Linear Regression.
- c) Classification: Classification derives a model to determine the class of an object based on its attributes. In other words, it classifies a data item into some of several predefined categorical classes. A collection of records will be available, each record with a set of attributes. One of the attributes will be class attribute and the goal of classification task is assigning a class attribute to the new set of records. The following algorithm are used for classification:
  - i. Decision tree
  - ii. Naive biased classification
  - iii. Generalized Linear Models (GLM)
  - iv. Super vector machine etc.
  - v. K-Nearest Neighbour (KNN)
  - vi. Random Forest
- d) Time Series Analysis: Time series is a sequence of events where the next event is determined by one or more of the preceding events. Time series reflects the process being measured and the components that affect the behaviour of the process. Time series analysis includes methods to analyze time-series data in order to extract useful patterns,

trends, rules and statistics. Stock market prediction is an important application of time-series analysis.

B. Descriptive data mining methods: This method is used to analyse the sequence and behaviours in the dataset. This type of methods are unsupervised learning methods. Descriptive data mining involves several techniques described as follows:

a) Clustering: In clustering technique, the data set is divided in various groups, known as clusters. As per clustering phenomenon, the data point of one cluster and should be more similar to other data points of same cluster and more dissimilar to data points of another cluster. There are two ways of initiating clustering algorithm: Start the clustering algorithm with no prior assumption and starting clustering algorithm with a prior postulate (Upadhyay and Katiyar., 2014). There various types of clustering algorithms include:

- i. K-Means Clustering
- ii. K-Medoids clustering
- iii. Hierarchical Clustering.
- iv. Grid Based Clustering.
- v. Density Based Clustering.
- vi. Optics Clustering.

b) Relationship mining: It is used for discovering relationships between variables in a dataset and encoding them as rules for later use. There are different types of relationship in mining techniques such as association rule mining (any relationships between variables), sequential pattern mining (temporal associations between variables), correlation mining (linear correlations between variables), and causal data mining (causal relationships between variables). In EDM, relationship mining is used to identify relationships between the student's on-line activities and the final marks and to model learner's problem solving activity sequences..

Association rule mining is an important data mining technique used to find an association between different datasets. The term is related to diverse tracking patterns and is helpful in creating groups of data that have dependently linked variables.

There are 2 core steps in this technique:

- i. Finding the frequently occurring data sets without missing any.

- ii. Creating strong association rules from the given frequent data sets.

Different types of association techniques are given below:

- i. Multilevel Association
- ii. Multidimensional Association
- iii. Quantitative Association

These techniques find useful applications in the retail industry where different patterns are identified to analyse the audience interest and how businesses should focus to boost up their profits.

- c) Discovery with Models: Its goal is to use a validated model of a phenomenon (using prediction, clustering, or knowledge engineering) as a component in further analysis such as prediction or relationship mining. It is used for example to identify the relationships between the student’s behaviour and characteristics.
- d) Outlier Detection: The goal of outlier detection is to discover data points that are significantly different than the rest of data. An outlier is a different observation (or measurement) that is usually larger or smaller than the other values in data. In EDM, outlier detection can be used to detect deviations in the learner’s or educator’s actions or behaviours, irregular learning processes, and for detecting students with learning difficulties (Upadhyay and Katiyar., 2014) (Baradwaj 2011)

Figure Table 2.1 shows categories of Educational Data Mining Methodology, their objectives and their key applications.

Table 2.1: Educational Data Mining Methodology Categories, (Algarni, 2016)

	Category	Objective	Key applications
1	Prediction	Develop a model to predict some variables base on other variables. The predictor variables can be constant or extract from the data set.	Identify at-risk students. Understand student educational outcomes.
2	Clustering	Groups specific amount of data to different clusters based on the characteristics of the data. The number of clusters can be different based on the model and the objectives of the clustering process.	Find similarities and differences between students or schools. Categorizes new student behavior.

3	Relationship Mining	Extract the relationship between two or more variables in the data set.	Finds the relationship between parent education level and students dropping out from school. Discovery of curricular associations in course sequences; Discovering which pedagogical strategies lead to more effective/robust learning.
4	Discovery with Models	It aims to develop a model of a phenomenon using clustering, prediction, or knowledge engineering, as a component in more comprehensive model of prediction or relationship mining.	Discovery of relationships between student behaviors, and student characteristics or contextual variables; Analysis of research question across wide variety of contexts.
5	Distillation of Data for Human Judgment	The main aim of this model to find a new way to enable researchers to identify or classify features in the data easily.	Human identification of patterns in student learning, behavior, or collaboration; labeling data for use in later development of prediction model.

### 2.1.1.7 Challenges of Educational Data Mining

The research on EDM from the year 1998 to 2012 found out that maximum research focuses were on academic objectives. The other issues are:

1. Educational data is incremental in nature: Due to the exponential growth of data, the maintaining the data in data warehouse is difficult. Monitoring the operational data sources, inferring the student interest, intentions and its impact in a particular institution is a major issue. Another issue is the alignment and translation of the incremental educational data.
2. Lack of Data Interoperability: Scalable Data management has become critical considering wide range of storage locations, data platform heterogeneity and a plethora of social networking sites, an example is Metadata Schema Registry which is a tool to enhance Meta data interoperability. There is also the need to design a model to classify/ cluster the data or find relationship between the data. Neuro-Fuzzy mining technique can be improved to remove the gap of data interoperability.
3. Possibility of Uncertainty: Due to the presence of uncertain errors, no model can predict hundred percent accurate results in terms of student modeling or overall academic planning.

4. **Research Expertise Relation between Student-Teacher.** In most of the higher Educational institutions, final year students have a compulsory project work which is a research work based on their area of interest. Generally Supervisors are assigned as per availability and area of expertise in the respective department. It is still not possible to assign all the students to supervisor with similar area of interest hence the result of the project is not applicable to real scenarios. There is need to find the relation between areas of interest, students' interest, applicability of the project/research and mining cross faculty interest. Association Mining can be introduced to optimize the issue (Jindal and Borah., 2013).

### 2.1.2 Intelligent Agents

Intelligent agents are entity that is able to act autonomously in a particular environment using sensors (input) and actuators (output) for achieving its goals. Intelligent agents have four components, the sensors, actuators, effectors and environment.

**Sensor:** Sensor is a device which detects the change in the environment and sends the information to other electronic devices. An agent observes its environment through sensors.

**Actuators:** Actuators are the component of machines that converts energy into motion. The actuators are only responsible for moving and controlling a system. An actuator can be an electric motor, gears, rails, etc.

**Effectors:** Effectors are the devices which affect the environment. Effectors can be legs, wheels, arms, fingers, wings, fins, and display screen.

**Environment:** The environment the agent will be working in

Figure 2.4 illustrates how agent interacts with its environment

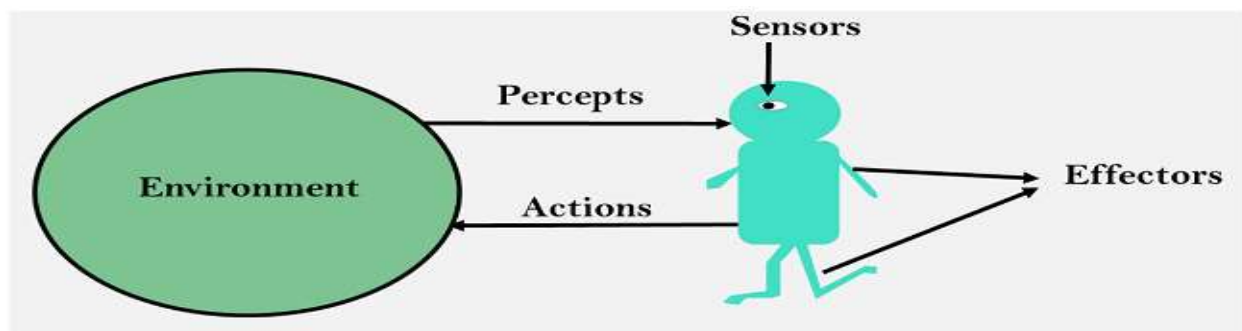


Figure 2.4 Intelligent agents and its environment (Bringsjord and Govindarajulu., 2018)



### **2.1.2.1 Features /Characteristics of Intelligent Agents**

Agent technology appears to be a promising solution to challenges of modern environment. This appears as a high level of software abstraction and it is a part of artificial intelligence. An agent can be defined as “An encapsulated computer system that is situated in some environment and that is capable of flexible, autonomous action in that environment in order to meet its design objectives (Wooldridge., 1995). Agent is a process which operates in the background and performs activities when specific events occur. The various properties of agents make them more suitable to environments where human intervention creates a great overhead. Agents are capable of relieving human intervention significantly and help in proper functioning of the system. The various characteristics of agents are:

- i. **Autonomy:** Autonomy corresponds to the independence of a party to act as it pleases. This characteristics means that an agent is able to act without direct intervention from humans or other agents, the agent has almost complete control over its own action and internal state.
- ii. **Proactive:** A proactive agent is one that can act without any external prompts. It acts in anticipation of the future goals.
- iii. **Reactive:** Agent responds based on the input it received and according to the environment. It responds in timely fashion to the environmental change.
- iv. **Communication:** It can be defined as those interactions that preserve the autonomy of the parties concerned.
- v. **Dynamism:** Agents are dynamic as their reaction is dynamic and varies according to the environment. (Srinivasan., 2015)
- vi. **Cooperation:** An agent should be able to interact with other agents. This can be arranged via Agent Communication Language (ACL)
- vii. **Sociability:** The agent should be capable of interacting in a peer to peer manner with other agents or human
- viii. **Adaptivity:** This agent characteristics means that the agent is capable of reacting flexibly to changes within its environment. It is able to accept goal directed initiatives when

appropriate and is also capable of learning from its own experiences, environment and interaction with other agents or human.

- ix. **Situatedness:** When an agent receives some form of sensory input from an environment, it then performs some actions that change its environment in the same way (Balaji and Srinivasan., 2010)

### **2.1.2.2 Multi-Agent Systems (MAS)**

A multi-agent system is a loosely coupled network of problem-solving entities (agents) that work together to find answers to problems that are beyond the individual capabilities or knowledge of a single entity (agent) (Sajja 2008). A multi-agent system is independent if each individual agent pursues its own goals independently of the others. A MAS is discrete if it is independent, and if the goals of the agents bear no relation to one another. Discrete MAS involve no cooperation. However agents can cooperate with no intention of doing so and if this is the case then the cooperation is emergent.

Multi-agents can solve complex issues effectively; such issues would have been too large for a single agent to solve. Agents can provide information as and when it is required and can handle the knowledge independently. Multi agents thus have a high applicability since the data from different sources are different. In a multi-agent system, there is a set of agents that work in their own sphere of influence, since the agents control different parts of the environment. If their spheres overlap, it may cause dependencies between the agents. The two principles on which the multi agents work are a high collaboration between the agents and a high degree of parallelism. The multi agents systems can be thus used when we have a complex problem to deal with, which can be broken down into sub parts, when a parallel approach will help save time, when a certain degree of redundancy is required and when the data comes from various sources and the data, controls, resources are all distributes across various system nodes (Fariz et al., 2015).

### **2.1.2.3 Types of Agents**

1. **User Interface Agent:** The user interface agent is responsible for communication with the user, which includes accepting the task to be performed as input and providing the results as output. It is responsible for inter-agent communication. User interface agents are used

to monitor the user interactions with the application and can control various aspects of that interaction.

2. **Agent manager:** On receiving a request from the interface agent, the agent manager forms a plan to complete the request. The agent manager is responsible for the completion of the user request, which it attains by assigning the work to different agents. The results are communicated to the interface agent. It is responsible for synchronization of the agents.
3. **Data mining agent:** A data mining agent is a software program built for pre-purpose of finding information efficiently. It is a type of intelligent agent that operates valuable information to find the relationship between different pieces of information. The mining agent initiates the mining technique based on the information provided by the knowledge module, such as the appropriate type of method for the problem at hand, the requirements of the method, form of input data, etc.
4. **Result agent:** The result agent receives the data mining result from the mining agent. The result agent is responsible for the presentation and visual representation of the knowledge with the help of the visualization primitives and report templates it maintains.
5. **Broker agent:** The broker agent contains the names, ontology and capabilities of all the agents registered with it. On receiving a request, the broker agent provides the corresponding names of the agents in order to fulfil the request.
6. **Query agent:** A query agent is created for each user request. The query agent uses the knowledge module schemas to generate queries in order to complete a user request.
7. **Filtering Agent:** Filtering agents, as their name implies, act as a filter that allows information of particular interest or relevance to users to get through, while eliminating the flow of useless or irrelevant information.
8. **Information Agent:** A parallel agent type to the filtering agent, which cuts down the information received, is the information agent, which goes out and actively finds information for the user. Information agents which are used primarily on the Internet and World Wide Web, can scan through online databases, document libraries, or through directories in search of documents that might be of interest to the user (Bigus 1996)
9. **Extraction Agent:** Extraction agent extracts set of information regarding object and is used its dilute for any such information for further needs. Any information fetch it

explain every criteria of objects. Collect complete information about the concept. It shows object detailed in well-mannered.

10. Retrieval Agent: Retrieval agent retrieve information which one has been extracting. Retrieval Agent executes information using data sets and visualization effects etc. It displays exactly induced information as well. Such functional values are used for retrieving procedure.
11. Office Agent: Office agent chooses information where it finds suitable. Different types of office agent occur; their work and uses are totally different depending on the functioning values
12. Workflow Agent: Work flow agent is an office management agent that automates daily tasks/routines that take up so much time at the office. These tasks include scheduling meetings, sending faxes, holding meeting review information, and updating process documents. The workflow agent can be configured for polling on demand processing.
13. System Agents: System agents are software agents whose main job is to manage the operations of a computing system or a data communications network. These agents monitor for device failures or systems overload and redirects work to other parts in order to maintain a level of performance and/or reliability.
14. Brokering or Commercial Agents: An agent that acts as a broker is a software program that takes a request from a buyer and searches for a set of possible sellers using the buyer's criteria for the item of interest. When the potential sellers are found to satisfy the request, the broker agent can return results to the user, who chooses a seller, and manually executes the transaction. The agent can also automatically execute the transaction on behalf of the user.

#### **2.1.2.4 Advantages of Multi-Agents**

Multi-agent systems have been widely adopted in many application domains because of the beneficial advantages offered. Some of the benefits available by using MAS technology in large systems are:

- i. Reduced cost: This is because individual agents cost much less than a centralized architecture.

- ii. Reusability: Agents have a modular structure and they can be easily replaced in other systems or be upgraded more easily than a monolithic system.
- iii. An increase in the speed and efficiency of the operation due to parallel computation and asynchronous operation.
- iv. A graceful degradation of the system when one or more of the agent fail. It thereby increases the reliability and robustness of the system.
- v. Scalability and flexibility- Agents can be added as and when necessary.

#### **2.1.2.5 Limitations of Multi-Agents**

Though multi-agent systems have features that are more beneficial than single agent systems, they also present some critical challenges. Some of the challenges include:

- a. Environment: In a multi-agent system, the action of an agent not only modifies its own environment but also that of its neighbours. This necessitates that each agent must predict the action of the other agents in order to decide the optimal action that would be goal directed. This type of concurrent learning could result in non-stable behaviour and can possibly cause chaos (Balaji & Srinivasan., 2010).
- b. Perception: In a distributed multi-agent system, the agents are scattered all over the environment. Each agent has a limited sensing capability because of the limited range and coverage of the sensors connected to it. This limits the view available to each of the agents in the environment. Therefore decisions based on the partial observations made by each of the agents could be sub-optimal and achieving a global solution by this means becomes intractable.
- c. Abstraction: In agent system, it is assumed that an agent knows its entire action space and mapping of the state space to action space could be done by experience. In MAS, every agent does not experience all of the states. To create a map, it must be able to learn from the experience of other agents with similar capabilities or decision making powers. In the case of cooperating agents with similar goals, this can be done easily by creating communication between the agents. In case of competing agents, it is not possible to share the information as each of the agents tries to increase its own chance of winning. It

is therefore essential to quantify how much of the local information and the capabilities each of the agents must know to create an improved modeling of the environment.

- d. Conflict resolution: Conflicts stem from the lack of global view available to each of the agents. An action selected by an agent to modify a specific internal state may be bad for another agent. Under these circumstances, information on the constraints, action preferences and goal priorities of agents must be shared between to improve cooperation. A major problem is knowing when to communicate such information and to which of the agents.
- e. Inference: A single agent system inference could be easily drawn by mapping the State Space/problem state to the Action Space based on trial and error methods. However in MAS, this is difficult as the environment is being modified by multiple agents that may or may not be interacting with each other. Further, the MAS might consist of heterogeneous agents have different goals and capabilities. Identifying a suitable inference mechanism in accordance of the capabilities of each agent is crucial in achieving global optimal solution (Balaji and Srinivasan., 2010)

It is not necessary to use multi-agent systems for all applications. Some specific application domains which may require interaction with different people or organizations having conflicting or common goals can be able to utilize the advantages presented by MAS in its design.

### **2.1.3 Agent Mining**

Agent mining refers to the application of autonomous intelligent agents in the field of data mining in order to support and enhance the knowledge discovery and decision making process while providing high performance and scalability. Due to their autonomous, flexible, mobile, adaptable and rational nature, agents are an excellent choice for parallel, multi-source, distributed mining. In agent driven data mining, for instance, agents can be used for data selection, data integration, data preprocessing, classification, clustering, association rules mining as well as knowledge presentation. In data mining for agents, data mining is used to extract knowledge from large datasets in the form of decision trees or data induces rules, which provide logic for the intelligent agents. For instance, consider an enterprise resource planning system that

maintains a log of all decisions and actions taken by a company. Using data mining, the developer can identify, code and encapsulate the logic behind these decisions and actions into agents that are robust and trustworthy enough to replace the human decision making process (Al-Barky and Ali., 2012).

The agent-data mining collaboration may occur and can be analysed in a number of diverse dimensions:

- Resource dimension at data, information, and knowledge levels.
- Infrastructure dimension at infrastructure, architecture, and process levels.
- Learning dimension at learning methods, learning capabilities, and performance levels.
- Interaction dimension for coordination, cooperation, negotiation, and communication.
- Social dimension in social and organizational factors for instance, in human roles.
- Performance dimension in the performance enhancement of one end of the coupled system.
- Interface dimension at the human-system interface, user modeling and interface design level.
- Application dimension in applications and domain problems (Al-Barky and Ali., 2012).

The integration of data mining and agents provides us with benefits concerning performance and simplicity, which paves the path for the use of more intelligent and complicated agent systems in order to attain more advanced goals. Data mining requires highly trained professionals to perform the multistep process from accessing and preparing data to presenting valuable knowledge to decision makers or executives. Agent mining provides for the automation of the mining steps enabling non-experts to use the system while assisting the work of experts too. Apart from providing high performance, the data mining process supported by agents helps to increase the quality of knowledge obtained, simplify the process of identifying patterns from huge data volumes as well as help in take good decisions in real time (Al-Barky and Ali., 2012).

#### **2.1.3.1 Data Mining using Multiple Agents**

Data-mining systems differ in certain ways from the machine learning algorithms which they are typically derived from. Firstly, they have to cope with large amounts of data. For example,

learning over a census database containing information on millions of families is very different from looking at a few hand-crafted examples of 'model' families. The second problem is that real world data has a tendency to contain errors and missing information. Finally, a data-mining system aims to discover knowledge that is novel, useful, and understandable, which typically requires a human to focus the search and to provide feedback on the knowledge discovered. One or more agents per network node are responsible for examining and analyzing a local data source. In addition, an agent may query a knowledge source for existing knowledge such as rules or predicate definitions. The agents communicate with each other during the discovery process. This allows the agents to integrate the new, individual knowledge they produce into a globally coherent theory. A user communicates with the agents via a user-interface. In addition, a supervisory agent, responsible for coordinating the discovery agents may exist. Figure 2.6 shows data mining using multi-agent system. The interface allows the user to assign agents to data sources, and to allocate high level discovery goals. It allows the user to critique new knowledge discovered by the agents, and to direct the agents to new discovery goals, including ones that might make use of the new knowledge.

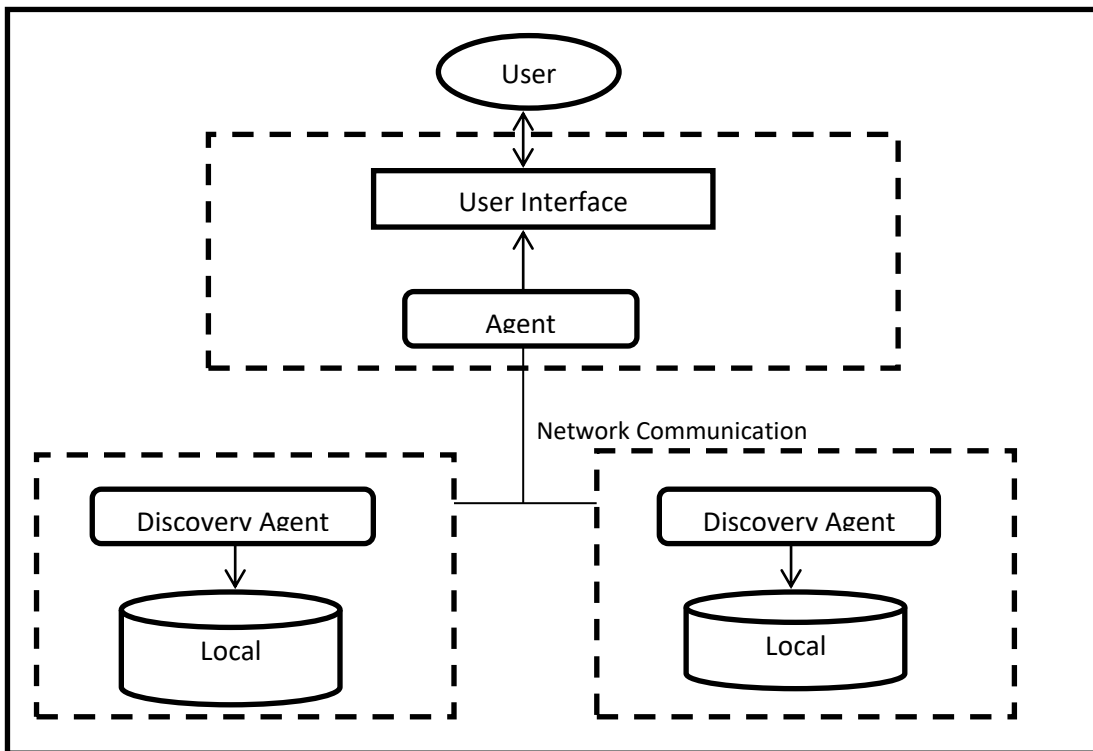


Figure 2.5: Data Mining Using Multiple Agents (Hemamalini & Josephine., 2014)



#### **2.1.4 Academic performance**

Academic performance is the extent to which a student, teacher or institution has achieved their short or long-term educational goals. Cumulative GPA and completion of educational benchmarks such as secondary school, diplomas, bachelor's degrees and postgraduate degree represent academic achievement.

Academic performance achievement is the level of achievement of the students' educational goal that can be measured and tested through examination, assessments and other form of measurements. Academic achievement is commonly measured through examinations or continuous assessments but there is no general agreement on how it is best evaluated or which aspects are most important: procedural knowledge such as skills or declarative knowledge such as facts (Ward et al.,1996). Furthermore, there are inconclusive results over which individual factors successfully predict academic performance, elements such as test anxiety, environment, motivation, and emotions require consideration when developing models of school achievement. An academic institution with more academic achievements is more likely to be sought than an institution with less achievement (Ziedner 1996).

##### **2.1.4.1 Student Academic Performance Determinant Attributes/Factors**

Understanding the factors (i.e., the predictor variables) that affect students' academic achievement is a critical input to understanding and improving the educational landscape (Sen and Ucar., 2012). Therefore, determining the variables that are related to academic achievement of students have always aroused the curiosity of researchers in educational data mining. Many of the previous studies analyzed this phenomenon one variable at a time. They tried to collect data, mostly from survey type instruments, to understand the relationship between a single factor and its impact on academic achievement. In literature, there are previous research works aimed at identifying the major factors or attributes that contributes in affecting the performance of students' and the methods that gives the best prediction result.

Shahiri et al., (2015) did a literature review on predicting students' performance by using data mining techniques. The work provided an overview of the data mining techniques that have been used to predict students' performance and how prediction algorithms can be used to identify the

most important attributes in students' data. According to their research the attributes that have been frequently used is cumulative grade point average (CGPA) and internal assessment. Through the coefficient correlation analysis, the result shows that CGPA is the most significant input variable by 0.87 compared to other variables. The main idea of why most of the researchers are using CGPA is because it has a tangible value for future educational and career mobility. Fadhilah et al., (2015) made use of student GPA in their works.

The research work of Pandey and Taruna (2014) considered 18 attributes from students' data of an engineering college and conducted an attribute selection measure to select the most important attributes. The most popular attributes selection measures are gain index, information gain and gain ratio for Decision Tree algorithms. The CGPA was the attribute with the highest gain ratio of 0.52861 and therefore the most important attribute in predicting performance of student. Some researchers studied the correlation between academic performance and parents educational background and income (Quadri and Kalyankar, 2010).

Ramesh et al., (2013) tried to identify the factors influencing the performance of students in final examination. They adopted survey cum experimental methodology to generate the database. The algorithms which were used by them for implementation were Naïve Bayes, Multi Layer Perception, SMO, J48, and REPTree. The results obtained from hypothesis testing reveals that type of school does not influence student performance but parent's occupation plays a major role in predicting grades. Bansode, (2016) previous academic performance and parent educational background are the most important attribute in predicting future academic performance of student.

Other research works focused on socio-economic status of student (Kolo et al., 2015). Some others investigated the impact of previous academic achievement in determining the performance of students in future. (Kolo et al., 2015; Bansode 2016) while others looked at how psychometric factors tend to affect the performance of students. (Ramaswami and Bhaskaran 2010; Sembiring et al., 2011 and Gray et al., 2014).

The next most important attribute being used is student demographic i.e gender, age, family background and disability. The reason why most of the researchers used students demographic such as gender was because they had different styles of female and male students in their learning process. (Bin et al., 2013).

#### 2.1.4.2 Student Performance Applications

A lot of researchers have shown a several practical applications on student performance pertaining to performance prediction, course recommendation, behaviour detection of students, career path planning and more.

- i. **Performance Prediction:** The performance prediction model can be built by applying data mining technique to an available collected data such as student CGPA / GPA and other variables .Previous work on student performance prediction used methods such as Bayesian Network, K-Nearest Neighbour, Decision Tree, Support Vector Machine, regression and others. Some of the models had categorized students into a few categories such as below satisfactory, satisfactory, and above satisfactory. Student performance can be predicted using student interaction with other students, instructors or teachers. By using performance prediction, underperforming students were identified providing them with relevant academic guidance to improve their study progress as well as final grade.
- ii. **Course Recommendation:** Previous studies have reported that student performance prediction will benefit institutions by improving learning process and course recommendation. Course recommendation can be proposed to student by analysing their previous result on CGPA or result on entry mode. By using course recommendation, it will identify course based on student qualification and interest. This recommendation will ensure that students are not misguided in choosing the field that are equivalent to their result and their interest. Asiah et al., (2019) researched on generated recommendation system that could give feedback and benefit on student performance. The subject or course recommendation was inspired by Austin Peay State University by developing subject recommendation system called ‘degree compass’ which pair current students with the best course that fit their talent and upcoming study program. The results from ‘degree compass’ algorithm successfully predicted more than 90% of subject accuracy.
- iii. **Detection of Students Behaviours:** Detection of undesirable student behavior aims to find out students who having some problem or unusual behavior such as: erroneous actions, low motivation, playing games, misuse, cheating, dropping out, academic failure, etc. Several soft computing techniques (predominantly classification and clustering) have been used to search and identify such students. Classification algorithms used for predicting, understanding and preventing academic failure includes decision tree, neural

networks, naïve Bayes, instance-based learning, logistic regression and support vector machines, feed-forward neural networks, probabilistic ensemble simplified fuzzy ARTMAP, Bayesian nets, logistic regression, simple logic classification, instance based classification, attribute selected classification, bagging, classification via regression, Bayesian classifiers, logistic models, rule-based learner, random forest, C4.5 decision tree algorithm, J48 decision tree algorithm, Farthest First clustering and algorithm, algorithm for the automatic identification of the students' cognitive styles.

Discriminant analysis, neural networks, random forests and decision trees have been used for classifying university students into low-risk, medium-risk and high risk of failing. Decision tree algorithms help earlier in identifying the dropouts and students who need special attention and allow teacher to provide appropriate advising/counselling

- iv. **Career Path Planning:** Studies have acknowledged that student academic performance had benefited higher educational institutions. Most of the researchers are using CGPA as a parameter to measure student performance. CGPA is a prominent value for future education and can be highlighted as a medium to determine potential candidate for job hunting as well as student's career mobility. Usually, the performance of CGPA will consider as an important criteria for student to complete their study and get back to career on time as planning. The student performance application also suitable for predicting student career and statistic information for relevant occupation and job seeking opportunity.
- v. **Course Scheduling:** Scheduling courses ("timetabling") at a University is a persistent challenge. Allocating course sections to prescribed "time slots" for courses requires advanced quantitative techniques, such as goal programming, and collecting a large amount of multi-criteria data at least six to eight months in advance of a semester. Bilal, (2017) designed a model to extract implicit knowledge from the higher education dataset, specifically the dataset concerns of the student satisfaction of courses-exams timetable using different and well-known classification algorithms.

#### 2.1.4.3 Algorithms for Predicting Student Performance

In educational data mining, predictive modeling is usually used in predicting students' performance. This is because a predictive model's main concern is the utility of the prediction

when applied to the unseen, (future cases). Predictive modeling, which is perhaps the most-used subfield of data mining, draws from statistics, machine learning, database techniques and optimization techniques. There are many success stories of predictive models mined from scientific data, financial data, and banking and insurance business data. Most of these reside in a large data warehouses. For some application area, maximizing accuracy or some other utility measure of the model is of paramount importance, even at the cost of giving up the ability to understand a simple model. For some applications, the quality of available data in terms of missing values and noise present extra challenges for model generation. (Hong and Weiss 2012). In order to build the predictive modeling, there are several tasks used, which are classification, regression and categorization.

The most popular task to predict students' performance is classification. Romero et al., (2008) in their research presented several mining algorithms that are used to mine educational data to classify students. There are several algorithms under classification task that have been applied to predict students' performance. Among the algorithms used are Decision tree, Artificial Neural Networks, Naive Bayes, K-Nearest Neighbor and Support Vector Machine.

i. **Decision Tree**

Decision Trees (DTs) are a non-parametric supervised learning method used for classification and regression. Decision trees learn from data to approximate a sine curve with a set of if-then-else decision rules. The deeper the tree, the more complex the decision rules and the fitter the model. Decision tree builds classification or regression models in the form of a tree structure. It breaks down a data set into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. A decision node has two or more branches. Leaf node represents a classification or decision. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data. Decision tree is one popular technique for prediction. Most of researchers have used this technique because of its simplicity and comprehensibility to uncover small or large data structure and predict the value (Shahiri et al., 2015). A comprehensive definition and characteristic of decision tree as it relates to predictive modeling was presented in the research work of Kabakchieva, (2013). The previous studies that adopt the Decision tree algorithm include; predicting drop out

features of students data for academic performance by Quadri et al., (2010), predicting third semester performance of MCA students by Mishra et al., (2014) and also predicting the suitable career for a student through their behavioral patterns. A decision tree algorithm was used by Blagojevic´&Micic., (2013) to predict the percentage of occurrence modules with selected input parameters in a learner’s management system of an e-learning platform.

The advantages of decision trees are that they represent rules which could easily be understood and interpreted by users, do not require complex data preparation, and perform well for numerical and categorical variables. Romero et al., (2008) said that the decision tree models are easily understood because of their reasoning process and can be directly converted into set of IF-THEN rules. Iterative Dichotomiser (ID3) and C4.5 are examples of decision tree algorithm

ii. **Neural Network**

Neural network is another popular technique used in educational data mining. Neural network is a set of connected input/output units and each connection has a weight present with it. During the learning phase, network learns by adjusting weights so as to be able to predict the correct class labels of the input tuples. Neural networks have the remarkable ability to derive meaning from complicated or imprecise data and can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. These are well suited for continuous valued inputs and outputs. Neural networks are best at identifying patterns or trends in data and well suited for prediction or forecasting needs (Baradwaj and Kumar., 2011).

The advantage of neural network is that it has the ability to detect all possible interactions between predictor’s variables (Gray, et al., 2014). Neural network could also do a complete detection without having any doubt even in complex nonlinear relationship between dependent and independent variables. Neural network technique is selected as one of the best prediction method. (Amirah et al. 2015).

iii. **Naive Bayes**

Naive Bayes algorithms assume that the effect that an attribute plays on a given class is independent of the values of other attributes. However, in practice, dependencies often exist among attributes (Kabakchieva 2013). In the research paper of Nikolovski et al.,

(2015) they explained that Naïve Bayes classifier is based on the Bayes rule of conditional probability. It analyzes all the contained attributes individually as though they are equally important and independent of each other. However, in practice, dependencies often exist among attributes; hence Bayesian networks are graphical models, which can describe joint conditional probability distributions. Naive Bayes algorithm is also an option for researchers to make a prediction. Several researchers have used Naive Bayes algorithms to estimate students' performance. The objective of their research is to find the most effective prediction technique in predicting students' performance by making comparisons. Their research showed that Naive Bayes has used all of attributes contained in the data. Then, it analyzed each one of them to show the importance and independency of each attributes (Osmanbegović et al., 2012).

The Naïve Bayes classifier technique is based on Bayesian theorem, whereas it performs better when data dimensionality is high (Nikam 2015). The Bayesian classifier is capable of calculating the most possible output based on the input. There is no problem to add new raw data at run time and have a better probabilistic classifier.

In this algorithm, the presence of a particular feature in a class is unrelated to the presence of any other feature. Describing by example why this algorithm is called a naïve, a fruit is judged as an apple when its characteristics are: round 3 inches in diameter and red, even it depends on each other or on other features, all of these properties independently contribute to the probability to judge that fruit is apple (Nikam2015).

Bayesian theorem provides an equation for calculating posterior probability  $P(c|x)$  from  $P(c)$ ,  $P(x)$  and  $P(x|c)$ :

$$p(c|x) = \frac{p(x|c)p(c)}{P(x)} \quad (2.1)$$

c: Class (target)

x: Predictor (attribute)

$P(c|x)$ : the posterior probability of class (c, target) given predictor (x, attributes).

$P(c)$ : the prior probability of class (Proposition)

$P(x|c)$ : the likelihood, which is the probability of predictor given class.

$P(x)$ : the prior probability of predictor (Evidence)

iv. **K-Nearest Neighbour (KNN)**

K-nearest neighbour is non-parametric lazy learner algorithm for classification and prediction. In order to classify a new instance, this algorithm checks the distance of its k neighbours from the training set to classify it. In general Euclidean Distance measure is used to find the distance. A training instance closest to the given test instance predicts the same class as this training instance. In WEKA this algorithm is available as IBK. (Pandey et al., 2014).

K-Nearest Neighbour method had taken less time to identify the students' performance as a slow learner, average learner, good learner and excellent learner.

K-Nearest Neighbour gives a good accuracy in estimating the detailed pattern for learner's progression in tertiary education (Gray et al., 2014).

K-nearest neighbor's algorithm is an instance-based learning algorithm that categorized objects based on closest feature space in the training set (Han *et al.*, 1999; Osuna, 2002). The training data is mapped into multi-dimensional feature space. The feature space is partitioned into regions based on the category of the training set. A point in the feature space is assigned to a particular category if it is the most frequent category among the k nearest training data. During the classifying stage, KNN classification approach finds the k closest labeled training samples for an unlabeled input sample and assigns the input sample to the category that appears most frequently within the k subset. As KNN outperforms the other classification approaches by its simplicity, it only requires a small training set with small number of training samples, an integer which specifies the variable of k and a metric to measure closeness (Osuna, 2002).

Some algorithms increase the speed of basic KNN algorithm e.g. ball tree, k-d tree, nearest feature line(NFL), tunable metric, orthogonal search tree and principal axis search tree. To more understand of KNN algorithm, suppose that an object is sampled with a set of different attributes, but the group to which the object belongs is unknown. Determining the class of a sample depends on evaluating the k-number of closest neighbours.

KNN classifies the test data using the training set directly. To classify any test data, it first calculates K value, which denotes the number of K-Nearest Neighbours. For each



test data, it calculates the distance between all the training data and then sorts the distance. Then by using majority voting, class label will be assigned to the test data.

The main objective of KNN is to find the nearest neighbour of an unknown data point and the value of k. If k=n, then the “n” nearest neighbour can be predicted. The final classification output is decided by calculating the distance between the test data and each of the training data with the help of KNN algorithm. The Euclidean distance between two points  $(x_1, y_1)$  and  $(x_2, y_2)$  in the plane is given by the equation

$$\text{Dist}((x_1, y_1) \text{ and } (x_2, y_2)) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (2.2)$$

v. **Support Vector Machine (SVM)**

Support Vector Machine is a supervised learning method used for classification. Support Vector Machines are another good technique for classifying educational data. The idea is to find data points (“support vectors”) which define the widest linear margin between two classes. Non-linear class boundaries can be handled by two tricks: first, the data can be mapped to a higher dimension, where the boundary is linear, and secondly, a soft margin can be defined, which allows some misclassification (Hamalainen et al., 2006) had chosen Support Vector Machine as their prediction technique because it suited well in small datasets. The only shortcoming is the “black-box” nature of the model. This is in contrast with the general requirement that models in intelligent tutoring system should be transparent. In addition, selecting appropriate kernel function and other parameters is difficult, and often we have to test different settings empirically.

Sembiring et al., (2011) stated that Support Vector Machine has a good generalization ability and faster than other methods. Meanwhile, the study done by Gray et al., (2014) demonstrated that Support Vector Machine method has acquired the highest prediction accuracy in identifying students at risk of failing.

Features of various classification algorithms with their strength and weaknesses are shown in table 2.2.

Table 2.2: Features and Limitations of Classification Algorithms (Nikam 2015)

SR.NO	ALGORITHM	FEATURES	LIMITATIONS
1	C4.5 Algorithm	<ul style="list-style-type: none"> <li>• Build Models can be easily interpreted.</li> <li>• Easy to implement.</li> <li>• Can use both discrete and continuous values.</li> <li>• Deals with noise.</li> </ul>	<ul style="list-style-type: none"> <li>• Small variation in data can lead to different decision trees.</li> <li>• Does not work very well on a small training data set.</li> <li>• Overfitting.</li> </ul>
2	ID3 Algorithm	<ul style="list-style-type: none"> <li>• It produces the more accuracy result than the C4.5 algorithm.</li> <li>• Detection rate is increase and space consumption is reduced.</li> </ul>	<ul style="list-style-type: none"> <li>• Requires large searching time.</li> <li>• Sometimes it may generate very long rules which are very hard to prune.</li> <li>• Requires large amount of memory to store tree.</li> </ul>
3	K-Nearest neighbor Algorithm	<ul style="list-style-type: none"> <li>• Classes need not be linearly separable.</li> <li>• Zero cost of the learning process.</li> <li>• Sometimes it is Robust with regard to noisy training data.</li> <li>• Well suited for multimodal classes.</li> </ul>	<ul style="list-style-type: none"> <li>• Time to find the nearest Neighbours in a large training data set can be excessive.</li> <li>• It is Sensitive to noisy or irrelevant attributes.</li> <li>• Performance of algorithm depends on the number of dimensions used.</li> </ul>
4	Naive Bayes Algorithm	<ul style="list-style-type: none"> <li>• Simple to implement.</li> <li>• Great Computational efficiency and classification rate</li> <li>• It predicts accurate results for most of the classification and prediction problems.</li> </ul>	<ul style="list-style-type: none"> <li>• The precision of algorithm decreases if the amount of data is less.</li> <li>• For obtaining good results it requires a very large number of records.</li> </ul>
5	Support vector machine Algorithm	<ul style="list-style-type: none"> <li>• High accuracy.</li> <li>• Work well even if data is not linearly separable in the base feature space.</li> </ul>	<ul style="list-style-type: none"> <li>• Speed and size requirement both in training and testing is more.</li> <li>• High complexity and extensive memory requirements for classification in many cases.</li> </ul>
6	Artificial Neural Network Algorithm	<ul style="list-style-type: none"> <li>• It is easy to use, with few parameters to adjust.</li> <li>• A neural network learns and reprogramming is not needed.</li> <li>• Easy to implement.</li> <li>• Applicable to a wide range of problems in real life.</li> </ul>	<ul style="list-style-type: none"> <li>• Requires high processing time if neural network is large.</li> <li>• Difficult to know how many neurons and layers are necessary.</li> <li>• Learning can be slow.</li> </ul>

(Kumar et al., (2017) did a comparative analysis of the accuracy of several classifiers in predicting student’s performance. Various attributes ranging from academic results, demographic factors and social factors were used for the analysis. From the result of the analysis in table 2.3 and 2.4, it was discovered that Naive Bayes had 100% accuracy and K-Nearest Neighbour algorithm also had 100%. For prediction of student academic performance, attributes like Gender, Age, Marital Status, Number of children, Occupation, Job associated with the computer, Bachelor, Another master, Computer literacy, Bachelor in informatics were used.

Table 2.3: Analysis of accuracy of classifier in predicting student’s performance (Kumal et al., 2017)

Author's	Attributes which affect prediction accuracy	DT	NB	RB	KNN	ANN
F Sarker et. al.	Internal attributes + students' first semester mark ( Model1)	--	--	--	--	74.5
F Sarker et. al.	Int + Ext attributes + students' first semester mark ( Model2)	--	--	--	--	76.5
F Ahmad et. al	Gender, race, hometown, GPA, family income, uni. mode entry, SPM grades	--	67.0	71.3	--	68.8
Mashael A.et. al.	first midterm exam (Predict Students Failure)	--	91	55.0	--	89.8
M Suljic et. al.	GPA, URK , MAT, VRI	--	76.6	--	--	73.9
R Asif et. al.	HSC, MPC and HSC marks, pre-uni marks, marks in different courses	73	83.6	56.9	74	67.6
El-Hakees et. al	SS_Type, HSC marks, City, Gender, Speciality	--	67.5	71.2	--	--
A Aziz et. al.	Gender, race, Hometown Location, Uni Entry Mode, Family Income	68.8	63.3	68.8	--	--
K D Kolo et.al.	status, gender	66.8	--	--	--	--
Jyoti Bansode	SSC marks, SSC medium, Admission type, mother's occupation	85	--	--	--	--
R. Sumitha et. al.	TWM, MOE, TOB, ATD ECUT, CGPA, arrears,	97.2	85.9	96.1	--	--
S V. Shinde et. al	Student's Internal Assessment	97.5	--	--	--	--
M Pandey et. al.	Academic information's, Demographic information	98.8	91.5	84.1	--	--
N Goga et. al.	family, PEP, EES, end of the first session result	99.9	--	96.7	--	--
G. S Josan et. al.	Sex, INS-High, TOB, MOI, TOS, PTUI, S-Area, Mob, Com-HM, Netacs, Int-GR, Atdn	69.7	65.1	--	--	--
M Koutina et. al.	Gender, Age, Marital Status, No of children, Occu., Job associated with PC, Bachelor, Another master, Comp literacy, Bachelor in informatics	68.5	100	90.9	100	--

From the above table, we find that Maria Koutina and Katia Lida Kermanidis, In his research found the 100% accuracy with Naive Bayes and K-Nearest Neighbor algorithm [24]. They represented their result in

Table 2.4: Summary of Student Academic Performance Prediction Techniques with Their Accuracy (Kumar et al., 2017)

Data Mining Techniques	Decision Tree	Naive Bayes	Rule Based	K-Nearest Neighbour	Nural Network
Highest Accuracy	99.9%	100%	96.7%	100%	89.8%
Lowest Accuracy	66.8%	63.3%	55.0%	74%	67.6%
Average Accuracy	83.35%	81.65%	75.85%	87%	78.7%

Fadilah et al., 2015 also did a comparative analysis of classification algorithm in predicting student’s performance. Attributes ranging from psychometric factors, GPA, demographic factors, high school background, social network interaction and other attributes were used in the analysis.

Decision tree classifiers, Support vector machine, Naive Bayes, Artificial Neural network, K-Nearest Neighbour were used in the analysis. Summary of the analysis is shown on the table 2.5

2.5: Attribute, Algorithm and data mining techniques frequently used to predict students' academic performance (Fadhilah et al., 2015)

S/N	Author	Attributes/Variables used	Algorithm	Algorithm Accuracy	Data mining technique
1	Ramaswami and Bhaskaran (2010)	Psychometric factors	Decision tree (CHAID)	CHAID 44.69%	statistical
2	Sembiring et al., (2011)	psychometric factors	Kernel k-means, smooth support vector machine (SSVM)	SSVM= 93.7%	Clustering and Classification
3	Osmanbegovic and Suljic, (2012)	CGPA, Student Demographic, High school background, Scholarship, Social network interaction	Naïve bayes, J48 Decision tree, Multi layer Perceptron	NB=76.65% MLP=71.93% J48=73.93%	Classification
4	Kabakchieva, (2013)	Place and profile of secondary school, final secondary education score, successful admission exam, the score achieved at that exam, and the total admission score	Decision tree, Naïve Bayes, K-Nearest Neighbour, Rule learners	J48=66.59% NB=<60% KNN=60% JRip= 63%	Classification
5	Ramesh et al.,(2013)	Student demographic, and secondary sch. background	Decision tree, Neural Network, Naïve Bayes	DT=65% NN=72% NB=50%	Classification
6	Tekin (2014)	GPA	Extreme learning machine, support vector machine and neural network	EML=94.92% SVM=97.98% NN=93.76%	Clustering and Classification

7	Gray et al.,(2014)	Previous academic record, demographic and Psychometric factors	Decision tree, Neural Network, K-nearest network, Naïve bayes, Support Vector Machine and Logistic regression	DT=65.93% NN=69.0% KNN 69.43% NB=68.03% SVM=73.33% LR=60.05%	Classification and Clustering
8	Asif et al., (2015)	Pre-university marks, GPA of first/second year	Naïve Bayes, Neural Network, Decision tree	NB= 83.65% NN=62.50% 1-NN =74.04%	Classification
9	Parneet et al., (2015)	Internal grade of student, Attendance count, Sex, Computer at home, internet access,type of secondary school	Multilayer Perception, Naïve Bayes, SMO, J48 and REPTree(decision tree)	MLP=75% NB=65.13% SMO=68.42% J48=69.73% REPTree=67.76%	Classification
10	Fadhilah et al., (2015)	GPA, Race, Gender, Family Income, University mode of entry	Decision tree, Naïve Bayes and Rule Based	RB=71.3% NB=67.0% DT=68.8%	Classification
11	Jishan et al., (2015)	CGPA, Midterm marks, Laboratory Marks, Attendance marks, Quiz Marks, Final grade	NB=75% NN=75% Note (NB is faster than NN)	NB=75% NN=75%	Classification, Optimal Equal Width Binning and Synthetic Minority Over-Sampling (SMOTE)

Table 2.5 continued: Table of Attribute, Algorithm and data mining techniques frequently used to predict students' academic performance (Fadhilah et al., 2015)

#### 2.1.4.4 Algorithms based parameters

The following are the algorithms based parameter:

- a. Accuracy: It is the ratio of number of correct predictions to the total number of input samples.
- b. Confusion Matrix: Table describing the performance of a classification model
- c. Probability Threshold: Presents the True Positive and True Negative rates.
- d. Execution Time: Time of running the algorithm on dataset.
- e. Precision: It is the number of correct positive results divided by the number of positive results predicted by the classifiers

- f. Recall: It is the number of correct positive results divided by the number of all relevant samples (all samples that should have been identified as positive).
- g. Receiver operating characteristics (ROC): A graph showing the performance of a classification model at all classification thresholds..
- h. F-measures: It is the Harmonic Mean between precision and recall. The range for F-measure Score is [0, 1]. It tells you how precise your classifier is (how many instances it classifies correctly), as well as how robust it is
- i. Mean Absolute Error: It is the average of the difference between the Original Values and the Predicted Values. It gives us the measure of how far the predictions were from the actual output. (Ashyaet al., 2018)

#### 2.1.4.5 Combining Classifiers

Several classifiers with a common objective are called multi-classifiers. In machine learning, multi-classifiers are sets of different classifiers which make estimates and are fused together to obtain a better result. They are referred to multi-classifiers, multi-models, multiple classifier systems, combining classifiers, decision committee etc (<https://quantdare.com/dream-team-combining-classifiers-2/>)

There are two ways to combine algorithm, namely:

1. Ensemble Method
2. Hybrid Method

1. **Ensemble Method:** This method allows the use of multiple learning algorithms to obtain better predictive results than you could have used one algorithm. It is a machine learning technique that combines several base models in order to produce one optimal predictive model. Ensemble methods falls under three categories: Bagging and Boosting

- a. Bagging (Bootstrap Aggregation) is a parallel process which involves training the same classifiers on different subsets of one data set. It is used to reduce variance and isn't recommended for models with high bias.

Multiple Classifiers are created from the original dataset. On each of these smaller datasets, a classifier is built, usually; the same classifier is built on all the datasets.

These models or classifiers run in parallel and are independent of each other. The final predictions are made by combining the predictors from all the classifiers through a majority voting scheme as shown in figure 2.6.

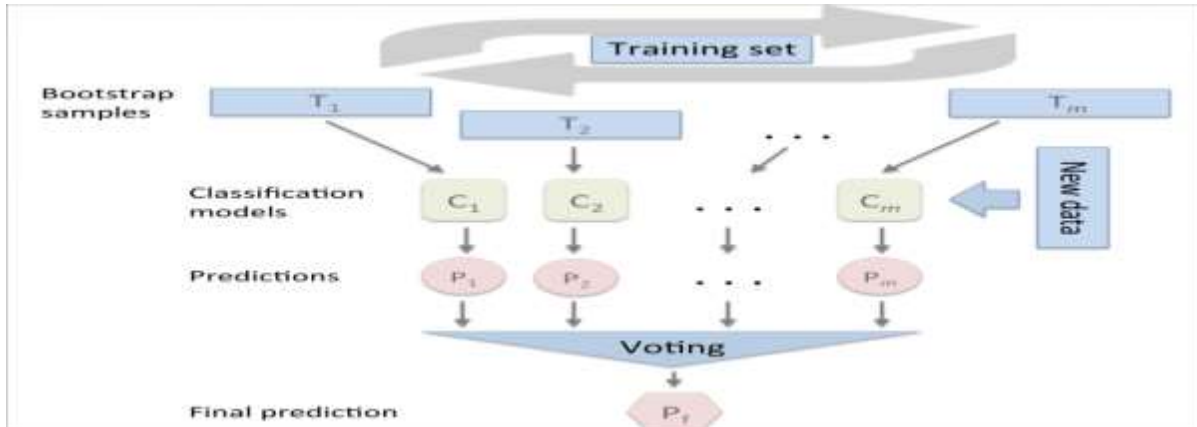
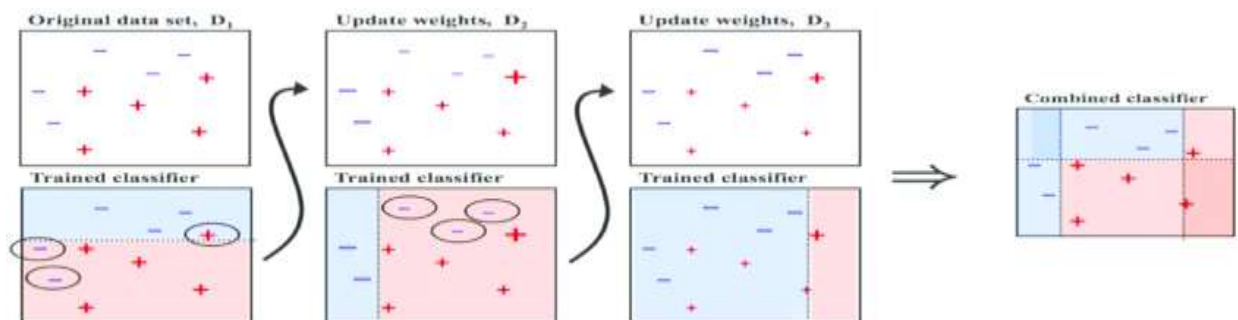


Figure 2.6: Bootstrapped Aggregate Ensemble Method

(<https://www.oreilly.com/library/view/python-deeper-insights/9781787128576/ch07s04.html>)

- b. **Boosting** is an ensemble modeling technique which attempts to build a strong classifier from the number of weak classifiers. It is done building a model by using weak models in series

First a model is built from the training data. Then a second model is built which tries to correct the error present from the first model (those that were incorrectly classified). The procedure is continued and classifiers are added until either the training data is predicted correctly or the maximum numbers of classifiers are added as shown in figure 2.7.



.Figure 2.7 Boosting Ensemble Method (<https://medium.com/swlh/boosting-and-bagging-explained-with-examples-5353a36eb78d>)

Boosting is used to reduce bias in an under fit model. Example of Boosting include Adaptive Boosting (ADABOOST), Gradient Boosting and Extreme Gradient Boosting (XGBoost)

Similarities and differences between boosting ensemble and bagging ensemble are highlighted in figure 2.6.

Table 2.6 Similarities and Difference between Boosting and Bagging (<https://quantdare.com/what-is-the-difference-between-bagging-and-boosting/>)

	Similarities	Differences
1	Both are ensemble methods to get N learners from 1 learner	While they are built independently, Boosting tries to add new models that do well where previous models fail.
2	Both generate several training data sets by random sampling	Boosting determines weights for the data to tip the scales in favor of the most difficult cases.
3	Both make the final decision by averaging the N learners (or taking the majority of them)	it is an equally weighted average for Bagging and a weighted average for Boosting, more weight to those with better performance on training data
4	Both are good at reducing variance and provide higher stability	Boosting tries to reduce bias. On the other hand, Bagging may solve the over-fitting problem, while Boosting can increase it.

2. **Hybrid Method:** Hybrid algorithms have been developed by combining two or more algorithms to improve or enhance overall search efficiency. Researchers often try to use the advantages of individual algorithms for the common good; at least that is the intention in principle. In practice, whether a hybrid can really achieve better performance is another matter, and finding ways to combine different algorithms to develop new algorithms is still an open problem.

In case of ensemble classifiers, multiple but homogeneous, weak models are combined, typically at the level of their individual output, using various merging methods, which can be grouped into fixed (e.g., majority voting), and trained combiners (e.g. Decision templates). Hybrid methods, in turn, take different, learners and combines them using new learning technique. Stacking is a main hybrid multi-classifier. (Avula &Asa., 2018)



Individual approach involves using a single statistical or machine learning technique for classification. The hybrid and ensemble models are efficient and robust because they combine the complementary features of more than one learning technique and overcome the weakness of individual techniques. Hybrid models can be stand-alone, transformational, tightly coupled or fully coupled. Hybrid models are of 4 types: Classification combined with Classification, Classification combined with Clustering, Clustering combined with Clustering and Clustering combined with Classification. Ensemble learning uses various base classifiers combined using a particular strategy of combination such as bagging, boosting, voting, etc. (Avula and Asa., 2018).

Before combining algorithms you need to know the strengths and weaknesses of each algorithm you are considering to combine together.

#### **2.1.4.6 Cross Validation**

Cross Validation is based on the principle that testing the algorithm on a new data set will yield a better estimate of its performance (Rafaeilzadeh et. al 2009). Using the same data set for the training and validation of an algorithm yields an overoptimistic result. Most real applications have a limited amount of data. As a result of this, the dataset is split into the training sample and the validation sample. The training sample is used to train the algorithm and the validation sample is used as “new data” to evaluate the performance of the algorithm. There are various methods of cross validation. They include holdout method, K-Fold Cross Validation and Leave-One-Out Cross Validation.

**2.1.4.6.1. Holdout Method:** This method is the simplest kind of cross validation in which the dataset is separated into two sets: the training set and the validation set. Using the training data only, the algorithm is trained. The algorithm is then used to evaluate the data in the validation set. The errors it makes are accumulated to give the mean absolute test set error, which is used to evaluate the algorithm. Nevertheless, since the evaluation may depend on which data is in the training set and in the validation set, its evaluation can have a high variance (Arlot&Celisse 2010).

**2.1.4.6.2. K-Fold Cross Validation:** This method is an improvement of the holdout method. In the k-fold cross validation method the dataset is divided into k subsets and the holdout method is repeated kth times. In each run, one of the k sets is used as the validation set and the remaining k-1 sets are put together to form the training set. Then the average error across all k trials is computed. This reduces the dependency of the evaluation on how the dataset is separated. Every data point is in the training set k times and in the validation set k-1 times. The variance in the result is reduced as k is increased. The disadvantage of this method is that it takes k times as much computation for an evaluation since the training algorithm has to be run k times. (Rafaeilzadeh et. al 2009).

#### **How to perform k fold cross validation**

- i. Shuffle the dataset randomly.
- ii. Split the dataset into **k** groups.
- iii. For each unique group: Take the group as a hold out or test data set. Take the remaining groups as a training data set. Fit a model on the training set and evaluate it on the test set.
- iv. Summarize the skill of the model using the sample of model evaluation scores.

**2.1.4.6.3. Leave-One-Out Cross Validation:** This method is a logical extreme of k-fold cross validation where k is equal to the number of data points. The training on the algorithm is done on all data points except for one. The evaluation given by the leave-one-out cross validation is good but computationally expensive

#### **2.1.4.7 Model Fitting in Machine Learning**

Fitting is a measure of how well a machine learning model generalizes to similar data to that on which it was trained. Generalization refers to the model's ability to adapt properly to new, previously unseen data drawn from the same distribution as the one used to create the model. Fitting is the essence of machine learning. If your model doesn't fit your data correctly, the outcomes it produces will not be accurate enough to be useful for practical decision-making. A properly fitted model has hyper parameters that capture the complex relationships between known variables and the target variable, allowing it to find relevant insights or make accurate predictions. Model fitting can be categorized into Overfitting and Underfitting.

**2.1.4.7.1 Underfitting** is when a machine learning algorithm cannot capture the underlying trend of data. It is just like trying to fit in an undersized pant. Its occurrence means that the model or algorithm does not fit the data well enough. It happens when there is less data to build an accurate model or when one tries to build a linear model with a non linear data. It is also often a result of an excessively simple model. Another cause of underfitting is where the model has “not learned enough” from the training data, resulting in low generalization and unreliable predictions. Underfitting occurs if the model or algorithm shows low variance but high bias. Underfitting can be avoided by using more data and also reducing the features by feature selection

**2.1.4.7.2 Overfitting:** A model is overfitted if we train it with a lot of data (just like fitting ourselves in oversized pants). When a model gets trained with so much data, it learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data. This means that the noise or random fluctuations in the training data is picked up and learned as concepts by the model. The problem is that these concepts do not apply to new data and negatively impact the model's ability to generalize. Overfitting occurs if the model or algorithm shows High variance but Low bias. There are two important techniques that you can use when evaluating machine learning algorithms to limit overfitting:

1. Use a resampling technique to estimate model accuracy.
2. Hold back a validation dataset. The most popular re-sampling technique is k-fold cross validation. It allows you to train and test your model k-times on different subsets of training data and build up an estimate of the performance of a machine learning model on unseen data.

**2.1.4.7.3 Good Fit:** Ideally, the case when the model makes the predictions with 0 errors is said to have a good fit on the data. This situation is achievable at a spot between overfitting and underfitting. Figure 2.8 shows a diagrammatic representation of underfitting, overfitting and appropriate fitting

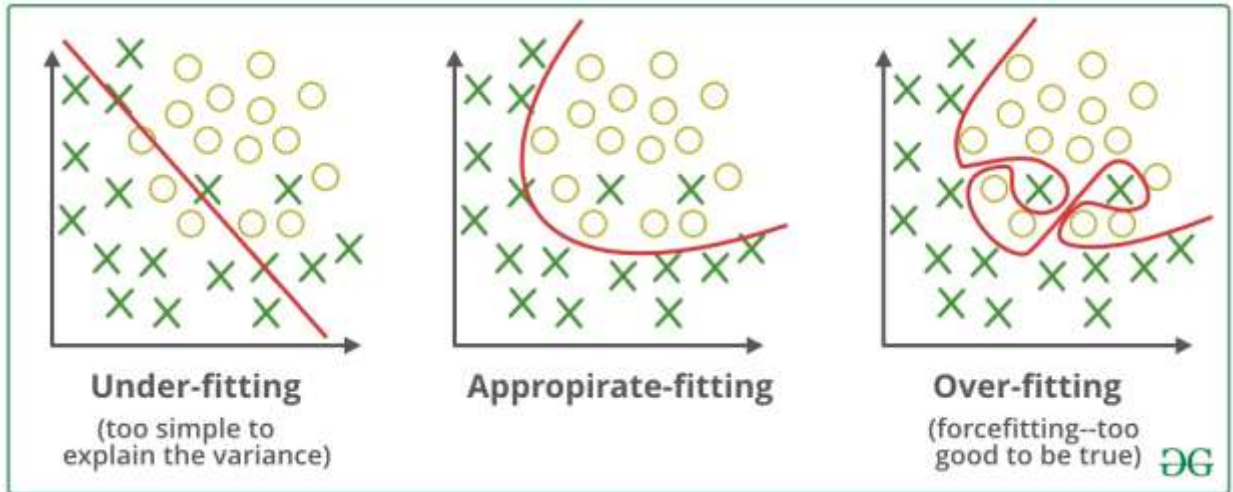


Figure 2.8 Overfitting, Underfitting and Appropriate Fit

(<https://www.geeksforgeeks.org/underfitting-and-overfitting-in-machine-learning/>)

#### 2.1.4.8 Data Mining Tools

Data mining tools and applications utilize machine learning algorithms, statistical analysis, artificial intelligence, and database systems. Data mining solutions serve as a means of processing high volumes of data. They aim at extracting the information required and then converting it into an understandable structure. Various data mining tools are explained hereunder:

##### 2.1.4.8.1 Waikato Environment for Knowledge Analysis (WEKA)

Waikato Environment for Knowledge Analysis (Weka) is a collection of machine learning algorithms for data mining tasks. These algorithms can either be applied directly to a data set or can be called from your own Java code. The Weka (pronounced Weh-Kuh) workbench contains a collection of several tools for visualization and algorithms for analytics of data and predictive modeling, together with graphical user interfaces for easy access to this functionality.

#### General Features

- i. Weka is a Java based open source tool data mining tool which is a collection of many data mining and machine learning algorithms, including pre-processing on data, classification, clustering, and association rule extraction

- ii. Weka provides three graphical user interfaces i.e. the Explorer for exploratory data analysis to support preprocessing, attribute selection, learning, visualization, the Experimenter that provides experimental environment for testing and evaluating machine learning algorithms, and the Knowledge Flow for new process model inspired interface for visual design of KDD process. A simple Command-line explorer which is a simple interface for typing commands is also provided by weka (Rangra, Bansal 2014) .

### **Advantages**

- i. Weka is best suited for mining association rules and it is Suited for machine Learning
- ii. It is also suitable for developing new machine learning schemes.(Witten and Frank., 2005)
- iii. Weka loads data file in formats of ARFF, CSV, C4.5, binary. Though it is open source, Free, Extensible, Can be integrated into other java packages.

### **Limitation**

- i. It lacks proper and adequate documentations and suffers from “Kitchen Sink Syndrome” where systems are updated constantly
- ii. Worse connectivity to Excel spreadsheet and non-Java based databases.
- iii. Weka is much weaker in classical statistics.
- iv. Does not have the facility to save parameters for scaling to apply to future datasets.

#### **2.1.4.8.2 Knowledge Extraction based on Evolutionary Learning (KEEL)**

KEEL is an application package of machine learning software tools. It is designed for providing solution to data mining problems and assessing evolutionary algorithms. It has a collection of libraries for pre-processing and post-processing techniques for data manipulating, soft-computing methods in knowledge of extracting and learning, and providing scientific and research methods.

### **Advantages**

- i. It includes regression, classification, clustering, and pattern mining and so on.
- ii. It contains a big collection of classical knowledge extraction algorithms, pre-processing techniques (instance selection, feature selection, discretization, imputation methods for missing values etc.), Computational Intelligence based learning algorithms, including

evolutionary rule learning algorithms based on different approaches (Pittsburgh, Michigan and IRL), and hybrid models such as genetic fuzzy systems, evolutionary neural networks etc.( Alcalá-Fdez et al., 2009)

### **Limitation:**

- i. Efficiency is restricted by the number of algorithms it support as compared to other tools.

**2.1.4.8.3 R Revolution** is a free software programming language and software environment for statistical computing and graphics. The R language is widely used among statisticians and data miners for developing statistical software and data analysis. One of R's strengths is the ease with which well-designed publication-quality plots can be produced, including mathematical symbols and formulae where needed.

### **General Features**

- i. The R project is a platform for the analysis, graphics and software development activities of data miners and related areas.
- ii. R is a well-supported, open source, command line driven, statistics package. There are hundreds of extra “packages” freely available, which provide all sorts of data mining, machine learning and statistical techniques. .
- iii. It allows statisticians to do very intricate and complicated analyses without knowing the blood and guts of computing systems

### **Advantages**

- i. Very extensive statistical library.
- ii. It is a powerful elegant array language in the tradition of APL, Mathematica and MATLAB, but also LISP/Scheme.
- iii. Ability to make a working machine learning program in just 40 lines of code
- iv. Numerical programming is better integrated in R
- v. R is more transparent since the Orange are wrapped C++ classes.
- vi. Easier to combine with other statistical calculations.
- vii. Import and export of data from spreadsheet is easier in R, spreadsheet are stored in a data frames that the different machine learning algorithms are operating on.

- viii. Programming in R really is very different, you are working on a higher abstraction level, but you do lose control over the details.

**Limitation:**

- i. Less specialized towards data mining.
- ii. There is a steep learning curve, unless you are familiar with array languages (Rangra&Bansal., 2014)

**2.1.4.8.4 Konstanz Information Miner (KNIME)** is an open source data analytics, reporting and integration platform. It has been used in pharmaceutical research, but is also used in other areas like CRM customer data analysis, business intelligence and financial data analysis. It is based on the Eclipse platform and, through its modular Application Programming Interface (API), and is easily extensible. Custom nodes and types can be implemented in KNIME within hours thus extending KNIME to comprehend and provide first tier support for highly domain-specific data format.

**General Features**

- i. Knime, pronounced “naim”, is a nicely designed data mining tool that runs inside the IBM’s Eclipse development environment.
- ii. It is a modular data exploration platform that enables the user to visually create data flows (often referred to as pipelines), selectively execute some or all analysis steps, and later investigate the results through interactive views on data and models.
- iii. The Knime base version already incorporates over 100 processing nodes for data I/O, preprocessing and cleansing, modeling, analysis and data mining as well as various interactive views, such as scatter plots, parallel coordinates and others.

**Advantages**

- i. It integrates all analysis modules of the well-known Weka data mining environment and additional plugins allow R-scripts to be run, offering access to a vast library of statistical routines. (Witten &Frank., 2005)
- ii. It is easy to try out because it requires no installation besides downloading and un-archiving.

- iii. The one aspect of KNIME that truly sets it apart from other data mining packages is its ability to interface with programs that allow for the visualization and analysis of molecular data.

**Limitations:**

- i. Have only limited error measurement methods.
- ii. Does not have automatic facility for Parameter optimization of machine learning/statistical methods. (Witten & Frank., 2005)

**2.1.4.8.5 RAPIDMINER** is a software platform developed by the company of the same name that provides an integrated environment for machine learning, data mining, text mining, predictive analytics and business analytics. It is used for business and industrial applications as well as for research, education, training, rapid prototyping, and application development and supports all steps of the data mining process. Rapid Miner uses a client/server model with the server offered as Software as a Service or on cloud infrastructures.

**General Features**

- i. Rapid miner is an environment for machine learning and data mining processes.
- ii. It represents a new approach to design even very complicated problems by using a modular operator concept which allows design of complex nested operator chains for huge number of learning problems.
- iii. Rapid miner uses Extensive Markup Language (XML) to describe the operator trees modeling knowledge discovery process.
- iv. It has flexible operators for data input and output file formats.
- v. It contains more than 100 learning schemes for regression classification and clustering analysis. (Mierswa et al., 2006).

**Advantages**

- i. Has the full facility for model evaluation using cross validation and independent validation sets.
- ii. Over 1,500 methods for data integration, data transformation, analysis and, modelling as well as visualization – no other solution on the market offers more procedures and therefore more possibilities of defining the optimal analysis processes



- iii. .RapidMiner offers numerous procedures, especially in the area of attribute selection and for outlier detection, which no other solution offers.

### **Limitations:**

Rapid Miner is the data mining software package that is most suited for people who are accustomed to working with database files, such as in academic settings or in business settings. The reason for this is that the software requires the ability to manipulate SQL statements and files. (Mikut&Wiley., 2011)

**2.1.4.8.6 ORANGE:** Orange is a component-based data mining and machine learning software suite, featuring a visual programming frontend for explorative data analysis and visualization, and Python bindings and libraries for scripting. It includes a set of components for data preprocessing, feature scoring and filtering, modeling, model evaluation, and exploration techniques. It is implemented in C++ and Python. Its graphical user interface builds upon the cross-platform framework (Rangra&Bansal., 2014)

### **General Features**

- i. Orange is a component-based data mining and machine learning software suite.
- ii. It includes a set of components for data preprocessing, feature scoring and filtering, modeling, model evaluation, and exploration techniques.
- iii. Data mining in Orange is done through visual programming or Python scripting.

### **Advantages**

- i. It is an open source data mining package build on Python, NumPy, wrapped C, and C++.
- ii. Works both as a script and with an extract, transform, load (ETL) work flow Graphic user interface (GUI).
- iii. Shortest script for doing training, cross validation, algorithms comparison and prediction.
- iv. Orange the easiest tool to learnm has better bugger.
- v. Scripting data mining categorization problems is simpler in Orange.
- vi. Orange does not give optimum performance for association rules.

### **Limitations**

- i. Not super polished.

- ii. The install is big since you need to install QT (A free and open source widget toolkit for creating graphic user interface)
- iii. Limited list of machine learning algorithms.
- iv. Machine learning is not handled uniformly between the different libraries.
- v. Orange is weak in classical statistics; although it can compute basic statistical properties of the data, it provides no widgets for statistical testing.
- vi. Reporting capabilities are limited to exporting visual representations of data models (Rangra&Bansal., 2014)

**2.1.4.8.7 SPSS Clementine** is a mature data mining toolkit which allows domain experts (normal users) to do their own data mining. Clementine has a visual programming which simplifies the data mining process. SPSS Clementine is one of the very first general purpose data mining tool.

**2.1.4.8.8 MATLAB** stands for matrix laboratory. It provides a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment. MATLAB include: math and computation, algorithm development, data acquisition, modeling, simulation, data analysis, exploration, and visualization.

**2.1.4.8.9 SAS Enterprise Miner** supports the entire data mining process. It streamlines the data mining process to create highly accurate predictive and descriptive models based on analysis of vast amounts of data from across an enterprise. SAS Enterprise miner provides the data access to relational and detail data stores. The base SAS language provides unrivalled power in aggregating and transforming data.

### **2.1.5.1 Academic Advising**

Academic Advising is a developmental process that assists students in the clarification of their life/career goal and in the development of Educational plans for the realization of these goals (Winston and Sandor., 1984). It is a decision-making process which assists students in realizing their maximum educational potential through communication and information exchanges with an advisor. It is ongoing, multi-faceted, and the responsibility of both student and advisor. The advisor serves as a facilitator of communication, a coordinator of learning experiences through

course and career planning and program process review, and an agent of referral to other campus services as necessary

#### **2.1.5.2 Factors Affecting Academic Advising**

Although research tends to show that there is a growing need for academic advising in institutions of learning, the programme may be hindered by a number of factors. In situations where advisors carry a heavy student-to-advisor load, the success of the programme may be limited. Institutional factors that affect the type of advising offered include large enrollment, type of programme, religious affiliation, institutional mission, and private or public status (Abelman et al., 2007).

Kennedy-Dudley (2007) found that senior students had a more positive evaluation of advising than their juniors. This may imply that they are in more need and may tend to seek academic advising more. A study on 350 students in post secondary institutions revealed that students in older classes paid more visits to their mentors. Students who are about to complete their studies would want to get information related to their future educational and career goals and therefore will be more likely to seek the advice of their academic mentors among other sources.

Gender has been identified as a factor that affects students' tendency to seek academic advising. Generally and traditionally, males have been less willing to seek help in dealing with academic difficulties (Daubman and Lehman, 1993; Ryan and Pintrich, 1997) and career counseling (Rochlen et al., 1999; Di Fabio and Bernaud, 2008). Such lower rates of help seeking among males transcend racial and national limits (Neighbors and Howard, 1987; Oliver et al., 2005). Men do not fail to seek help because they do not have problems but because social norms of traditional masculinity frowns on help seeking by men (Kessler et al., 1981; Wisch et al., 1995; Lee, 1997; Möller-Leimkühler, 2002). Kennedy-Dudley (2007) found that women were more likely than men to have been advised professionally.

In a study by the National Science Foundation (2008), it was found that female respondents at the bachelor, master's and doctoral degree programme levels considered all types of mentoring roles to be significantly more important than male respondents. The exception to this trend was the Academic/Career factor, which showed no significant differences in gender for the masters' level respondents. These findings imply that gender is likely to influence perceptions on academic advising and the tendency to seek the service. In another study of 238 students (Clark

et al., 2005), it was reported that females had a higher perception of being mentored. Male students have less social support in university settings and are less likely to reach out for educational support (Hernandez et al., 2004). These findings imply that gender should be one of the factors to consider when planning for academic advising with the possibility of instituting an “intrusive” form of mentoring (Redmond, 1990) for male students. In intrusive mentoring, the advisor takes the lead and contacts the student on a periodic basis rather than waiting for him or her to initiate contact.

The availability of academic advisors is crucial for the success of the student-advisory programme, especially in colleges and universities. At the university, students may fail to make contact with their advisors due to their own tendency to leave immediately after class, lack of extracurricular involvement, the lack of on-campus residence, lack of on-campus employment, and the large number of adjunct instructors that do not have office hours (King, 1993). All these factors may hinder interaction between students and their mentors.

Many advisors bear additional responsibilities to advising students, including teaching, marking, performing committee work, working at institutional events, and undertaking various other duties that take time away from direct advising with students. Institutional duties may differ from institution to another, thus allowing plenty of time to advise for some academic mentors while leaving others with very little time for advising

### **2.1.5.3 Effects of Academic Advising**

In institutions where guidance counselors are overburdened and individualized attention is not always the norm, advisors play a critical role in answering questions, writing recommendation letters, and ensuring that students are on track to graduate. Malone (2009) noted that advising is key to students’ success. According to him, high school students need diverse support to gain skills and knowledge necessary to succeed in college including academic content competencies, college application guidance, cognitive and critical thinking skills, civic awareness, time management and teamwork strategies, and healthy social-emotional coping abilities.

In the view of Poliner and Lieber (2004), students’ academic skills can grow through academic advisory which a structured programme is built into the school/college day through which an adult and a small group of students meet regularly for academic guidance and support. Advisory programmes aim to lower individual students’ barriers to success.

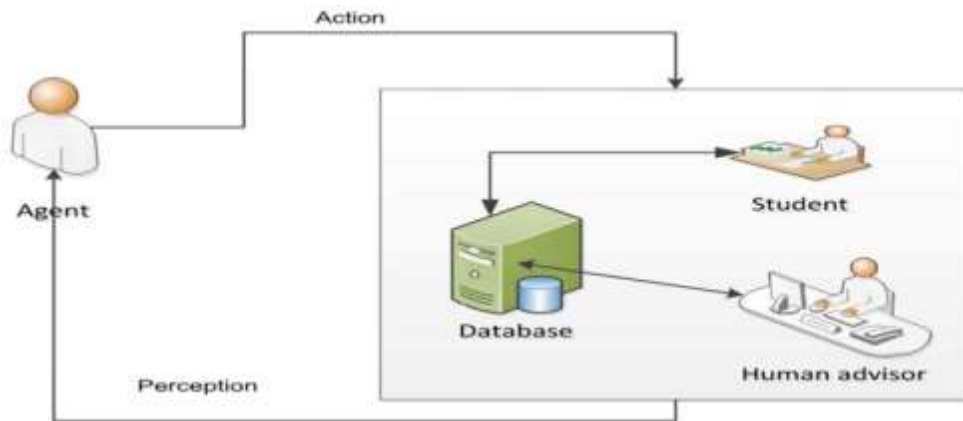
Mentoring is seen as a means for promoting student retention (Walker and Taub, 2001), particularly the retention of first-year students (Johnson, 2008). Research findings suggest that academic advising improves retention (McArthur, 2005; Sayles, 2005) (McLaren, 2004) through improved academic performance among other benefits. Research findings also indicate that mentoring has a positive impact on the personal and professional development of young adults (Levinson, 1978).

According to Habley (2004), one of the primary factors affecting college retention is the quality of interaction a student has with a concerned person on campus. Hester (2008) found that students who had increased interactions with their advisors had higher grade point averages (GPAs). In a study of 69 freshman students by Haught et al, (1998), it was found that students who received academic advising had a higher semester GPA at the end of the semester, and a higher cumulative GPA at the end of the following semester as compared to a control group. These findings imply that students who utilize advisors will benefit the most from the advising relationship.

A study by Pargett (2011) reported a positive relationship between academic advising and student development and student satisfaction with college. Students who are satisfied with college life are likely to be adjusted and focused as a result of which they may do well in their studies.

#### **2.1.5.4 Agent Oriented System in an Academic Advising System**

Academic Advising System (AAS) could be expressed in terms of agent-based system. Agreeably, there is no exact definition for an agent; nevertheless, there are some that are remarkable and accepted across many domains. One of the most widely accepted definition is given as follows: “An agent is an entity which perceives its environment and is able to act, typically autonomously and pro-actively, in order to solve particular problems, whilst remaining responsive to its environment. Relating the above definition to an academic advising environment, intelligent advising software would qualify as an agent, whereas its environment would include students, human advisors, databases, etc as illustrated in figure 2.9 below (Wilson 2018).



**Figure 2.9 Agent Relationship in an academic advising system. (Wilson 2018)**

### **2.1.5.5 Feature Selection and Attribute Ranking**

Feature selection is a preprocessing step to machine learning which is effective in reducing dimensionality, removing irrelevant data, increasing learning accuracy, and improving result comprehensibility. In some cases, accuracy on future classification can be improved; in others, the result is a more compact, easily interpreted representation of the target concept. There are three main categories wrapper, filter and embedded algorithms. The filter model selects some features without the help of any learning algorithm. The wrapper model uses some predetermined learning algorithm to find out the relevant features and test them. Wrapper model is more expensive than filter model because it requires more computations; hence filter method is preferred if there is large number of features. The commonly used ranking methods in different datasets include:

- i. Information Gain (IG) attribute evaluation:
- ii. Gain Ratio (GR) attribute evaluation,
- iii. Symmetrical Uncertainty (SU) attribute evaluation,
- iv. Relief-F (RF) attribute evaluation:
- v. Chi-Squared (CS) attribute evaluation
- vi. Filter evaluation

## 2.2 Review of Related work

Amin et.al (2017) proposed and validated a predictive GPA model by using machine learning approaches. A relatively small-sample experiment was used for determining the set of self-regulatory learning behaviors. For every constituent of the generated model, the predictability was quantified and its relevance was calculated. The major objective of grade prediction was utilizing the constructed models for designing the intervention strategies that help students when the academic failure is at risk. A probabilistic predictive model of GPA was used to define and detect the helpful interventions as per the mathematical calculations. The basic interventions were defined and the interventions which are of help to students having minimum GPA were identified by the application framework. 53% of accuracy was achieved by the proposed algorithm.

Merchán, et.al (2016) proposed a predictive model to be applied to predict academic performance of students. Several data mining methods were applied on the data of 932 students of a university in Columbia to evaluate and analyze their performances. On the basis of input data given, the expected results and output characterization and other factors, the evaluation of results achieved was done. The prediction accuracy was an important parameter to evaluate the performances as well. Considering the specific details of the population examined and the requirements specified by the institution, the said pertinence was evaluated. In preventing, academic risk and desertion, timely decisions were considered important along with the accompaniment of students with their learning procedure.

Sivasakthi, (2017) proposed a knowledge flow model using all the five different classifiers. Within the programming education field, the importance of prediction and classification based algorithms was also studied. For predicting the programming performance of students, five supervised data mining algorithms were applied on the data set. On the basis of predictive accuracy, the performance evaluations of these algorithms were done. It was seen that 93% accuracy was achieved in case of implementing Multilayer perceptron (MLP). Further, WEKA scenario was implemented to compare all the five classifiers. The performance of MLP was shown to be the best. The students that are very new to the introductory programming were identified through the research so that they can be helped with special attention.

Nakayama, et.al (2018) evaluated the regression models using the fitness models and analyzed their contributions. With the help of these evaluations, the effectiveness of learner's reflections was measured such that the learning performance was predicted. A variable selection technique was used to examine the contributing variables using a step-wise procedure. The R-squared ( $R^2$ ) and Akaike information criterion (AIC) indices were used to perform comparisons against the fitness of these models. When employment of indices of participant's reflection is applied, the improvement in performance of regression models is done. A variable selection technique was utilized to choose few reflection indices for the regression model even though the scores of final exams and change of variables were not in correlation. It thus proved that the hypothesis which states that the learning performance is affected by the contribution of assessment of reflections is correct.

Fan Yanga, et.al (2018) analyzed the performance of students, their progress and potentials using the multiple analysis tools. Initially, Student Attribute Matrix (SAM) was used to formulate the student model along with performance and non-performance related attributes. Secondly, BP-NN algorithm was applied for providing student performance estimation tools. The prior knowledge of students and their performance attributes were used to estimate the attributes of students. For describing the progress of students related to different aspects along with the casual relationships, the BP-NN was used to propose the student progress indicators and attributes. The level at which a factor would affect the performance of student was known by the indicators and predictor. It was thus possible to train up the students. The real academic performance data which was gathered from 60 high school students was used to check the performance of the analysis tools. Correct and highly accurate results were achieved by applying the proposed tools as per the evaluation results.

Paris et al., (2010) compared the data mining methods accuracy of classifying students to predict class grade of a student. The predictions were more useful for identifying the weak students and assisting administration to take remedial measures at initial stages to produce excellent graduate that will graduate at least with the second class upper.

Minaei-Bidgoli and Punch (2001) applied data mining classifiers as a means of comparing and analyzing students' use and performance for those who had taken a technical course via the web.



The results showed that combining multiple classifiers led to a significant accuracy improvement in a given data set.

Devasia et al., (2016) proposed classification within the information of student such that on the basis of previously existing information, the division of students can be predicted. Naïve theorem was applied since several techniques were used for knowledge classification within the area unit. For the prediction of performance at the top of that particular semester, various types of information were collected from the previous information of the students available. Students who needed special guidance can be highlighted through the study.

Dewan Md. Farid et al., (2014) anticipated a hybrid algorithm using decision tree and naive bayes algorithm. The method was mainly aimed at increasing the accuracy of classification for multi class-class classification tasks. The proposed hybrid algorithm showed the better sensitivity, specificity, cross validation and classification accuracy on real benchmark data sets. The proposed hybrid algorithm showed 90% accuracy.

Ulyani et.al (2017) presented that the major factor that leaves a huge impact on the behavioral intentions of student is the service quality performance. Questionnaires were distributed within seven Malaysian public and private universities. The descriptive statistics and covariance-based structural equation modeling were used to analyze the data. The least likely execution of favorable behavioral intentions was influenced by the freedom, serenity, management dimensions as well as aesthetic factors. A positive behavior towards the student housing was seen as per the results achieved when students adapted to live in multi-cultural community in which they would have access to good hospitality, personal privacy and appropriate building ambiance.

Safri et al., (2018) applied a combination of Naive Bayes Classifier algorithm and K-Nearest Neighbor determining the feasibility of healthy Indonesian card recipients; the Naive Bayes Classifier algorithm aimed to minimize variation within an attribute to obtain accurate higher than the K-Nearest Neighbor algorithm alone.

Bakar et al., (2008) proposed an agent based data classification approach, and it was based on creating agent within the main classification process. They showed the use of agent within the classification theory, which would help to improve classification speed, and maintain the quality of knowledge. The proposed agents were embedded within the standard rule application

techniques. The result showed the significant improvements in classification time and the number of matched rules with comparable classification accuracy

Rathee and Mathur (2013) applied ID3, C4.5 and CART decision tree algorithms on the educational data for predicting a student's performance in the examination. All the algorithms were applied on the internal assessment data of the student to predict their academic performance in the final exam. The efficiency of various decision tree algorithms was analyzed based on their accuracy and time taken to derive the tree. The predictions obtained from the system helped the tutor to identify the weak students and improve their performance. C4.5 was the best algorithm among all the three because it provided better accuracy and efficiency than the other algorithms.

Amrieh et al., (2016) studied on performance prediction at the University of Jordan. A data set of students from different countries was used. In addition to using individual machine learning methods, the researchers also applied ensemble methods, and compared the results between them. Decision trees provided the best results. Another area that the researchers focused on was behavioural features. It was found that the inclusion of behavioural features improved the prediction results (Amrieh et al., 2016).

Naren (2014) proposed a system that specifies the classification techniques for predicting the career options and to predict the violent behaviour prevalent among students. A response sheet was used to gather details regarding the background information, reaction of a student when irritated and the interests of a particular student. These parameters were mined to find the corresponding behavioural pattern of a student. However, additional research on student attitudes and learning was needed to predict the patterns efficiently. Minimal number of attributes was used which in turn did not give accurate values. Only limited amount of information was suggested for prediction of career options for students.

Nitya & Vinodini (2014) proposed a system for predicting the academic performance and the behavior of a student. Various techniques such as Naïve Bayesian Classification, Multilayer Perceptron, J48 and ID3 were used to analyze the attributes. However only a comparative analysis on which technique would give accurate results was specified but not on how they needed to be implemented.

Acharya and Sinha (2014) applied Machine Learning Algorithms for the prediction of students' results. They found that best results were obtained with the decision tree class of algorithms.

Kotsiantis, et al, (2004) applied five classification algorithms namely Decision Trees, Perceptron-based Learning, Bayesian Nets, Instance-Based Learning and Rule learning to predict the performance of computer science students from distance learning stream of Hellenic Open University, Greece. A total of 365 student records comprising several demographic variables like sex, age and marital status were used. In addition, the performance attribute namely "mark in a given assignment" was used as input to a binary (pass/fail) classifier. Filter based variable selection technique was used to select highly influencing variables and all the above five classification models were constructed. It was noticed that the Naïve-Bayes algorithm yielded high predictive accuracy of 74% for two-class (pass/fail) dataset.

Arsad, et. al (2013) applied the method of Artificial Neural Network (ANN) for the prediction of academic performance of students. The cumulative grade points (CGPA) were used as the measuring criterion. Data was collected from electrical department of Teknologi MARA University, Malaysia. The first semester result of students was taken as the input predictor variable (Independent variable) and eighth semester grade points are taken as the output variable (Dependent variable). The study was done for two different entry points namely; Matriculation and Diploma intakes. Performances of the models were measured using the coefficient of Correlation R and Mean Square Error (MSE). The outcomes from the study showed that fundamental subjects at semester one and three had strong influence in the final CGPA.

Huda et. al (2017) presented two prediction models for the estimation of student's performance in final examination. The work made use of 395 data samples provided by the University of Minho in Portugal, which focused on the performance in math subject. To ensure better comparison, Support Vector Machine algorithm and K-Nearest Neighbour algorithm were applied to the dataset to predict the student's grade. Empirical studies outcome indicated that Support Vector Machine achieved slightly better results with correlation coefficient of 0.96, while the K-Nearest Neighbor achieved correlation coefficient of 0.95.

Mohan et. al (2015) used two type of techniques for the overall prediction of the students' performance over a huge volume of data. Those techniques were Learning Analytics and

Predictive Analytics. The required data for building the prediction model was collected from the CBSE schools, MySQL server was also used for storing the huge amount of data. Pre-processing step was done by using apache HIVE framework. From the pre-processed data needed information were discovered using MapReduce algorithm in the Hadoop framework. The Predictive analytics part where the actual predictions were made was done using the multiple linear regression model.

Cortez and Silva (2008) attempted to predict failure in the two core classes (Mathematics and Portuguese) of two secondary school students from the Alentejo region of Portugal by utilizing 29 predictive variables. Four data mining algorithms ; Decision Tree (DT), Random Forest (RF), Neural Network (NN) and Support Vector Machine (SVM) were applied on a data set of 788 students, who appeared in 2006 examination. It was reported that DT and NN algorithms had the predictive accuracy of 93% and 91% for two-class dataset (pass/fail) respectively. It was also reported that both DT and NN algorithms had the predictive accuracy of 72% for a four class dataset.

Osmanbegovic and Suljic (2012) applied three supervised algorithms such as Decision Tree, Naïve Bayes, Neural Network based algorithms to predict the student grades in the upcoming examinations and learning accuracy of the students. They implemented the Decision tree algorithm for prediction of student performance such as grades or CGPA.

Arockiam et al., (2010) used FP Tree and K-means clustering technique for finding the similarity between urban and rural students programming skills. FP Tree mining was applied to sieve the patterns from the dataset. K-means clustering was used to determine the programming skills of the students. The study clearly indicated that the rural and the urban students differ in their programming skills. The huge proportions of urban students were good in programming skill compared to rural students.

Kuyoro, et al., (2013) published optimal algorithm suitable for predicting student's academic performance. He designed a framework of intelligent recommender system that could predict students' performance as well as recommend necessary actions to be taken to aid the students and identify background factors that affect students' academic performance in tertiary institution at the end of first year. Research used ten classification models and a multilayer perception; an

artificial neural network function generated using Waikato Environment for Knowledge Analysis (WEKA). The work shows that identifying the relevant student background factors could be incorporated to design a framework that could serve as valuable tool in predicting student performance as well as recommend the necessary intervention strategies to adopt.

Nguyen, et al., (2010) proposed a novel approach which uses recommender system techniques for educational data mining, especially for predicting student performance. They compare recommender system techniques with traditional regression methods such as logistic/ linear regression by using educational data for intelligent tutoring systems. Experimental results show that the proposed approach could improve prediction results.

Tsai and Chen (2010) explained that hybrid models are of four types: Classification combined with Classification, Classification combined with Clustering, Clustering combined with Clustering and Clustering combined with Classification. Ensemble learning uses various base classifiers combined using a particular strategy of combination such as bagging, boosting, voting, etc.

Brijesh &Saurabh., (2013) applied Bayesian classification on the student database from the higher education stage. The study aimed at identifying those students who needed more attention to reduce the drop out ratio and take action at a right time which helped to improve the performance of the students and the instructors.

Ragab et al., (2012) applied a hybrid procedure that was based on the data mining techniques and rules for admission. The System predicted the study track of the students with their profiles and validated it using the auditing process.

Kalles and Pierrakeas (2004) discussed different machine learning techniques (decision trees, neural networks, Naive Bayes, instance-based learning, logistic regression and support vector machines) and compared them with genetic algorithm based induction of decision trees. They discussed why the approach had a potential of developing into an alert tool. They embarked in an effort to analyze students' academic performance through the academic years, as measured by the students home work assignments, attempted to derive short rules that explain and predict success or failure in the final exams.

Moucary, et al., (2011) applied a hybrid technique on K-Means Clustering and Artificial Neural Network for students who are pursuing higher education while adopting a new foreign language as a means of instruction and communication. Firstly, Neural Network was used to predict the student's performance and then fitting them in a particular cluster which was formed using the K-Means algorithm. This clustering served as a powerful tool to the instructors to identify student capabilities during their early stages of academics.

Bhardwaj and Pal (2011) justified the capabilities of data mining techniques in context of higher education. Decision tree was used to evaluate students' performance at the end of semester. Variables considered were previous semester marks, class test grade, seminar, assignment, general proficiency, attendance, lab work, and end semester marks. The classification task used was able to predict the student division on the basis of the previous database. This helped to reduce failure ratio because early identification enable appropriate action.

In another study, Bhardwaj and Pal (2011) focused on using Bayesian classification algorithm to predict students' performance in BCA Dept of Indian Universities. Variables considered were Sex, Category, medium of teaching, student food habit, other habit, living condition, accommodation, family size, family status, family annual income, grade in senior secondary school, students' college type, father's qualification, mother's qualification, father's occupation, mother's occupation and grade obtained in BCA. Naïve Bayes classification algorithm was used as a technique to design the student performance prediction model. It is found that grade in senior secondary school, living condition, medium of teaching, mother's qualification, student other habit, family income and family status were high potential variable for student performance. The investigation shows that other factors outside students' effort have significant influence over students' performance.

Kabra and Bichkar., (2011) collected data from S.G.R. College of Engineering and Management, Maharashtra. They collected data from 346 students of engineering first year. Evaluation was performed using J48 algorithm by 10 fold cross validation. The accuracy of J48 algorithm was 60.46%. This model was successful in identifying the students who are likely to fail.

García-Saiz and Zorrilla (2011) collected student's data from University of Cantabria. They created two different datasets each with 194 records. They have used J48, Rtree, JRip, OneR and

NB classification models and proposed Meta algorithm. When they performed experiment using first data set, JRip algorithm had the highest 81.95% accuracy and J48 had the TPR of 95.62. When they performed experiment using second data set, J48 has the highest accuracy of 87.11% and TPR was high (98.83) for JRip algorithm. They compared all these results with proposed Meta algorithm. It was observed after comparison that proposed algorithm gave better results than other algorithms.

Shahiri et al., (2015) provided an overview on the data mining techniques that have been used to predict student's performance. The paper also focused on how the prediction algorithm can be used to identify the most important attributes in a student's data. The meta-analysis was based on the highest accuracy of prediction methods and also the main important factors that may influence the student's performance. Prediction accuracy that used classification method grouped by algorithms for predicting student's performance since 2002 to 2015 were presented. Neural Network had the highest prediction accuracy with 98% followed by Decision Tree by (91%). Next, Support Vector Machine and K Nearest Neighbor gave the same accuracy, (83%). Lastly, the method that has lower prediction accuracy is Naive Bayes by (76%).

Elakia and Aarthi (2014) designed a system to justify that various data mining techniques such as classification could be used in educational databases to suggest career options for the high school students and also to predict the potentially violent behaviour among the students by including extra parameters other than academic details. Three decision trees ID3, C4.5 and CHAID were compared for their various performance measures.

Ramesh et al., (2013) tried to identify the factors influencing the performance of students in final examination. They adopted survey cum experimental methodology to generate the database. The algorithms which were used by them for implementation were Naïve Bayes, Neural Network and Decision Tree. The attributes used included Students demographic, Students Secondary School background. Naïve Bayes had an accuracy of 50%, Neural Network: 72% and Decision Tree: 65%. The results obtained from hypothesis testing reveals that type of school does not influence student performance but parent's occupation plays a major role in predicting grades.

Tekins (2014) implemented several prediction techniques in data mining such as Extreme Learning machine, Support Vector Machine and Neural Network to assist educational

institutions with predicting their students' GPAs at graduation. Students who were predicted to have low GPAs at graduation were known earlier, then extra efforts were made to improve their academic performance and, in turn, GPAs. The attribute used was their CGPAs. From the result, Extreme Machine learning had an accuracy of 94.92%, Support Vector machine: 97.98%, Neural Network: 93.76%.

Gray et al., (2014) used personality, motivation and learning strategies variables gathered between the year 2010-2012 alongside six different classification algorithms (Decision Tree, Neural Network, K-Nearest Neighbour, Naïve Bayes, Support vector Machine and Logistic Regression) to predict student learning progression and achievement. The attributes used were previous academic record, demographic and psychometric factors. Decision tree: 65.93%, Neural Network: 69.0%, KNN: 69.43%, Naïve Bayes: 68.03%, SVM: 73.33% and Logistic Regression: 60.05%. The result from these studies showed there were strong correlation between the variables examined and performance of the student.

Asif et al., (2015) used data mining methods (Naïve Bayes, Neural Network and Decision Tree) to study the performance of undergraduate students. Two aspects of students' performance were focused upon. First, predicting students' academic achievement at the end of a four year study programme and second, studying typical progressions and combining them with prediction results. The attributes used were Pre-university marks and GPA of first/second year. Naïve Bayes had an accuracy of 83.65%, Neural Network: 62.50% and 1-NN: 74.04%. The results showed that by focusing on a small number of courses that are indicators of particularly good or poor performance, it is possible to provide timely warning and support to low achieving students, and advice and opportunities to high performing students.

Parneet et al., (2015) focused on identifying the slow learners among students by a predictive data mining model using classification based algorithms. Real World data set from a high school was taken and filtration of desired potential variables was done using WEKA, an Open Source Tool. The dataset of student academic records was tested and applied on various classification algorithms such as Multilayer Perception, Naïve Bayes, SMO, J48 and REPTree using WEKA. MLP had an accuracy of 75%, Naïve Bayes 65.1%, SMO: 68.42%, J48: 69.73%, REPTree: 67.6%.



Fadhilah et al., 2015 proposed a framework for predicting students' academic performance of first year bachelor degree students in Computer Science course. The data were collected from 8 year period of intakes from July 2006/2007 until July 2013/2014 containing the students' demographics, previous academic records, and family background information. Decision Tree, Naïve Bayes, and Rule Based classification techniques are applied to the students' data in order to produce the best students' academic performance prediction model. Rule Based had an accuracy of 71.3%, Naïve Bayes: 67% and Decision Tree: 68.8%.

Badr et al., (2016) built an application to predict student's performance in a programming course based on their previous performances in specific mathematics and English courses. In addition, the model aimed to reduce dropout rates by helping students predict their performance in programming courses before enrolling for them. Two experiments were conducted using the CBA rule-generation algorithm. They first used student's grades in two English courses and two mathematics courses, which generated four rules with accuracy of 62.75%. The second used student's grades only in two English courses, generating four rules with accuracy of 67.33%. The results showed that student's performance in English courses had a significant predictive effect on their performance in the programming course

Yadav et al, (2012) focused on generating predictive models for student retention management using decision tree algorithms (ID3, C4.5 and ADT) in WEKA. Study showed that intervention programs can have significant effects on retention, especially for the first year. Machine learning algorithms were applied to analyze and extract information from existing student data to establish predictive models. The predictive models were then used to identify among new incoming first year students those who are most likely to benefit from the support of the student retention program. The empirical results showed that short but accurate prediction list for the student retention purpose could be produced by applying the predictive models to the records of incoming new students. The study identified students who needed special attention to reduce drop-out rate.

Bhullar and Kaur (2012) took data set of 1892 students from various colleges for student performance prediction and evaluation. J48 algorithm was chosen for evaluation using 10 fold cross validation. Success rate of J48 algorithm was 77.74%.

Pandey and Sharma (2013) compared J48, Simple Cart, Reptree and NB tree algorithms for predicting performance of engineering students. They took data of 524 students for 10 fold cross validation and 178 students for percentage split method. It was discovered that J48 decision tree algorithm achieved an accuracy of 80.15% using 10 fold cross validation method and 82.58% using percentage split method.

Altujjar et al., (2016) built a predictive model based on records of female students using the ID3 decision tree algorithm to reveal the courses affecting low academic performance at the IT department, King Saud University, Riyadh, Saudi Arabia. They built several models based on ID3 decision tree algorithm. They divided the dataset into three groups to build separate model for each group. Results suggested that the classification model based on performance in the second year was the most accurate. The student performance in IT 221 and the two programming courses, CSC111 and CSC113, was a great indicator of student level of achievement.

Undavia et al., (2013) conducted analysis on data of 128 students of MBA and MCA using J48, simple CART and Random tree decision tree algorithms. Under test option 10 folds cross validation was used for implementation. They compared performance of the algorithms using student's data. After implementation it was observed that J48 and simple CART performed best in terms of accuracy and both had higher accuracy of 68.75% than random tree algorithm. J48 took less time in building a model than CART. This model worked well in predicting the students who had highest grade.

Aziz et al., (2014) conducted analysis on 399 records of students using naïve bayes, rule based and J48 decision tree algorithm. They used cross validation and percentage split method for evaluation. In cross validation 3, 5, 10 fold cross validation was performed and in percentage split method training: testing 10:90, 20:80, 30:70, 40:60, 50:50, 40:60, 30:70, 20:80, 10:90 percentage split were used. After comparison of the three classification algorithms, it was found that rule based and J48 decision tree algorithm had higher accuracy 68.8%.

Jadhav and Channe (2016) conducted two experiments to predict the student's final mark. The first experiment compared classification algorithms using three datasets in which a better result was achieved with better accuracy when all available data were taken into account versus filtering. In the second experiment, they concluded that the best accuracy can be obtained by applying classification model for both numerical and categorical data.

### 2.3 Summary of Literature and Knowledge Gap

From the Review of Literature, it was discovered that mostly used data mining algorithm for Predicting Students Academic Performance include Decision Tree (DT), Naive Bayes (NB), Artificial Neural Networks (ANN), Rule-based (RB) and K-Nearest Neighbour (KNN), Support Vector Machine (SVM).

Providing better education will need a lot of parameters to understand the process upon level of student understanding. A lot of researchers used predictive technique and tools to discover hidden characteristic to minimize the failure rate among the student.

For the purpose of performance prediction, important attributes such as previous records of the student are gathered; subsequently data mining techniques and classification algorithms are used to get a deeper insight and prediction.

Academic performance is affected by various factors which include psychometric factors, demographic factors, work related factors, social factors etc

From the literatures reviewed, the knowledge gaps include:

Inability for researchers to find useful indicator as well as parameter for the recommendation to enclose the evaluating analytics results in practice. Another challenge in predicting student academic performance is selecting the right factor and relevant attributes with a correct prediction method. To choose an appropriate method, most of the researchers applied mixed method approach to integrate best prediction technique to increase the robustness of the model. However, selecting an appropriate method is also depends on the availability of student data input for the model to perform the calculation for accurate prediction. The researchers focused on undergraduate students; there was no analysis for postgraduate studies. Another challenges concerned were about the small size of data due to incomplete and missing values.

In our study we hybridized two major kinds of classification techniques KNN and Naïve Bayes to predict the performance of Postgraduate Students, and improve results.

Therefore, the contributions of this research work can be viewed as:

- Identifying and Analysis of the highly influencing factors that affect student's academic performance.

- The research focuses on implementation and evaluation of classification techniques such as K-Nearest neighbour, Naïve Bayes and a Hybrid of KNN and Naïve Bayes. The result is then compared with the single classifiers in terms of accuracy, precision and speed.
- The results of the experiments will be verified based on the students' data combined with the previous semester results.

## CHAPTER THREE

### SYSTEM ANALYSIS AND METHODOLOGY

This chapter highlights the system analysis methods and research methodology used in the course of this study.

#### 3.1. System Analysis

System Analysis is a problem-solving technique that breaks down a system into its component pieces for the purpose of studying how well those component parts work and interact to accomplish their purpose.

It attempts to appraise the existing system and refine the abstract description of the project. Before any project is carried out, the system has to be analyzed to suit the description of how and what the system would be doing. The analysis can be on paper for clarity in a sequence and also explicit in a way a layman would easily understand. The system after analysis has to be designed following the series of steps from the formal description of that system.

The objective of analysis is a realistic and keen insight into a system and its problem areas, so that an improved system can be designed.

##### 3.1.1. Analysis of the Existing System

Two systems were analyzed: K-Nearest Neighbour (KNN) and Naïve Bayes Classifier

###### 3.1.1.1 KNN Classifier

Ihusan and Ashraf (2017) predicted students' performance using the KNN Classifier. The basis of nearest neighbor is the categorization of unknown data point in which its class is already known. Here, it is said to be non-parametric because it does not make assumptions on the underlying data distribution, that is, the model structure is determined by the data.

The nearest neighbor is calculated according to K-value that determines the number of nearest neighbor to be considered, hence, defines the class of a sample data point. It works by finding the distances between a query and all the examples (k) closest to the query, then votes for the most frequent label (in the case of classification).

The algorithm for K-nearest Neighbour is shown below. Depending on the function to be carried out, classification uses the mode of the K labels while regression calculates the mean of the K label as shown in figure 3.1 and 3.2 below

### **The KNN Algorithm**

1. Load the data
2. Initialize K to your chosen number of neighbors
3. For each example in the data
  - 3.1 Calculate the distance between the query example and the current example from the data.
  - 3.2 Add the distance and the index of the example to an ordered collection
4. Sort the ordered collection of distances and indices from smallest to largest (in ascending order) by the distances
5. Pick the first K entries from the sorted collection
6. Get the labels of the selected K entries
7. If regression, return the mean of the K labels
8. If classification, return the mode of the K labels

Figure 3.1 Algorithm for KNN Classifier (<https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm>)

```
k ← the number of nearest neighbor
for each object Z do
    Calculate the distance between every object x and x in the training set d(x,z)
    Neighborhood ← the k neighbors, closest to Z in the training set
    Z Class ← Select Class (according to neighbourhood)
End for
```

Figure 3.2: Pseudocode for KNN Classification Algorithm (Ihusan and Ashraf 2017)

### **3.1.1.1.2 Advantages of K-NN Classifier**

1. KNN algorithm is very simple to understand and equally easy to implement.
2. It has good predictive power
3. It is an instance-based learning; K-NN is a memory-based approach. The classifier immediately adapts as new training data is collected. It allows the algorithm to respond quickly to changes in the input during real-time use.
4. It performs well in both binary and multi class prediction
5. It requires zero to little training time before making predictions hence new data can be added seamlessly which will not impact the accuracy of the algorithm.
6. KNN might take some time while selecting the first hyper parameter but after that rest of the parameters are aligned to it.

### **3.1.1.1.3 Disadvantages of K-NN**

Despite the advantages offered by K-NN, it still exhibits some disadvantages:

1. It is very sensitive to outliers as it simply chose the neighbors based on distance criteria.
2. It has no capability of dealing with missing value problem.
3. The prediction time is quite high as it finds the distance between every data point.
4. It is not cost effective in that it takes a lot of memory to run (instances)

### **3.1.1.2 Naïve Bayes Algorithm**

Dake and Gyimah (2017) analyzed students' grades using naïve Bayes Classifier It is based on Bayesian theorem and performs better when the data dimensionality is high. The Bayesian classifier is capable of calculating the most possible output based on the input. Addition of new raw data at runtime gives a better probabilistic classifier. Here the presence of a particular feature in a class is unrelated to the presence of other features. In Naïve bayes, the likelihood of each class is calculated in the training set and the value with the greatest likelihood becomes the predicted value as shown in the algorithm in figure 3.3

Input:

Training dataset T,

F= (f1, f2, f3,...,fn) // value of the predictor variable in testing dataset.

Output:

A class of testing dataset.

Step:

1. Read the training dataset T;
2. Calculate the mean and standard deviation of the predictor variables in each class;
3. Repeat  
Calculate the probability of  $f_i$  using the gauss density equation in each class;  
Until the probability of all predictor variables (f1, f2,f3,..., fn) has been calculated.
4. Calculate the likelihood for each class;
5. Get the greatest likelihood;

Figure 3.3 Algorithm for Naïve Bayes Classifier

#### **3.1.1.2.1 Advantages of Naïve Bayes**

1. It is very simple, easy to implement and fast
2. If its conditional independence holds, then it will converge quicker than discriminative models like logistic regression
3. It needs less training data
4. It is highly scalable. It scales linearly with number of predictions and data points
5. It handles continuous and discrete data
6. It performs well in both binary and multi class

#### **3.1.1.2.2 Disadvantages of Naïve Bayes**

1. The classifier makes strong assumption on the shape of the data distribution, that is, any two features are independent, given the output class.



2. Due to data scarcity, for any possible value of a feature, the likelihood value is needed to be estimated by a frequent approach. This can result in probabilities going towards 0 or 1., thus leading to numerical instabilities

### 3.1.1.3 Existing Academic Performance Prediction Systems

In this research work, two existing academic performance prediction system using Naïve Bayes and KNN Classifies were analyzed

#### 3.1.1.3.1 Students Grades Predictor using Naïve Bayes Classifier

Dake and Gyimah (2017) proposed a Naïve Bayes approach in predicting student’s final grade. The Classifier model was built on the data of previous students who offered the same course. The attributes/features used included Attendance, Assignment, Test Score, Class participation, Hostel Proximity, Gender and Academic Standing. The attributes and their values were selected based on the discretion that can have effects on the students ability to pass or fail an examination.

To run the classifier, a total of 50 instances were taken for analysis. The comma-separated values (CSV) were converted to the Weka Attribute-Relation File Format (ARFF) using the ARFF-View. The training data set was then subjected to Naïve Bayes Classification. The result obtained gave 88% accuracy for correctly classified instances. The model was evaluated using the 10 folds validation. Out of the 50 instances, 44 were correctly classified and 6 were incorrectly classified. The students’ academic status was then predicted. The diagram in figure 3.4 highlights the stages for the classification student’s performance, Data is first collected from their various sources and relevant attributes or features are selected. Next pre processing is done. In this stage, data is transformed to the format which the classification can be done. It involves feature extraction, normalization and discretization. Naïve Bays Classifier is then applied to mine the patterns and discover important features in the data. Finally the results are evaluated

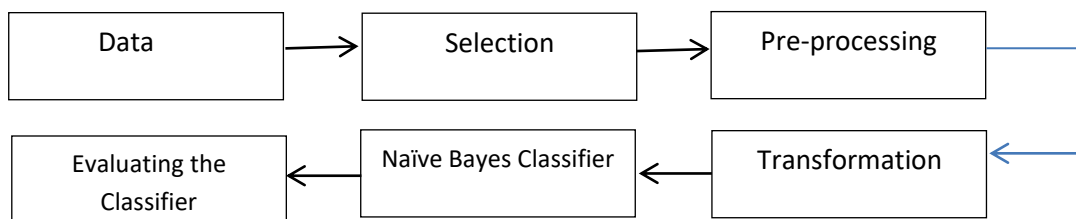


Figure 3.4 Extraction of Knowledge using Naïve Bayes Classifier (Dake and Gyimah 2017)

### 3.1.1.3.2 Predicating Academic Performance of Students in Higher Institution with KNN Classifier

Omisore and Azeez (2016) developed a predictive model using KNN Classification to predict the academic performance of students in higher institutions. Educational data set of 310 students from all levels in 2013/2014 session was collected from the University of Lagos, Akoka were used. Other relevant information was collected via questionnaires. Various selected features were used for the prediction. Variables/students attribute used include student’s demographic, current and previous academic standing, departmental structure and family background. Subsequently, data captured into different tables were joined and attributes with lesser entropy were removed as shown in figure 3.5 below. Students academic standing was stratified into five different groups- Distinctive, Good, Average, Weak and Hapless. 10 Folds Cross Validation was used to evaluate the performance. The result obtained gave a prediction accuracy of 58.3% . .

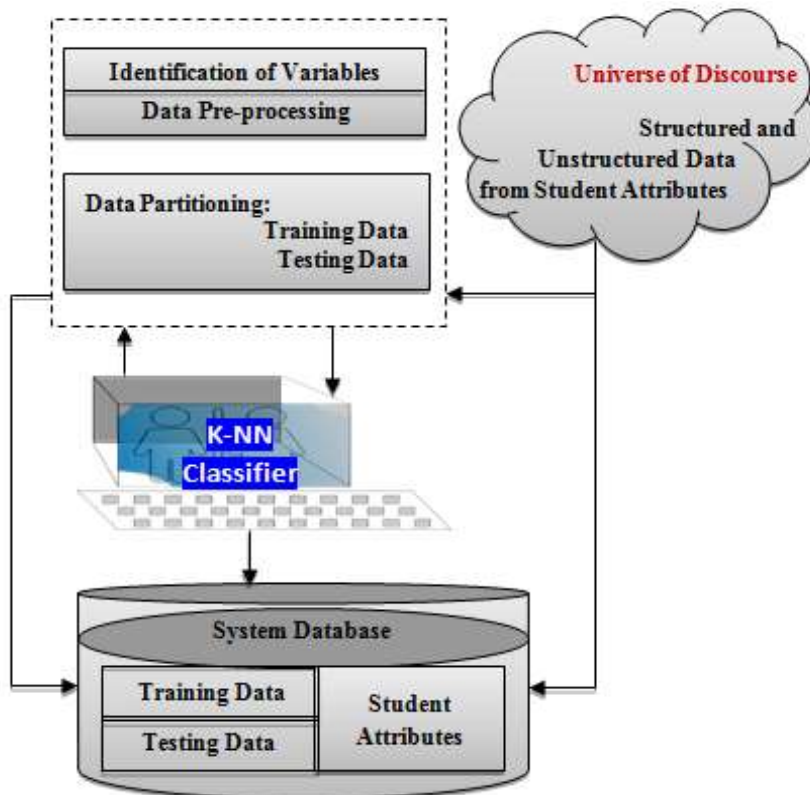


Figure 3.5 Conceptualized Model for Prediction of Students Academic Performance (Omisore and Azeez 2016)

### **3.1.1.3.3 Advantages of the Existing Prediction Systems**

1. The Model helped to classify new students according to their performance. It also helped the teachers give early advise students to improve their performance
2. Naïve Bayes and KNN are easy and fast to predict class of test data set. This reduced the calculation time. They also perform well in both single and multi-class prediction.
3. The attributes chosen were highly influential to improve the prediction accuracy of unknown classes
4. Naïve Bayes need less training data while KNN requires no training before making predications

### **3.1.1.3.4 Disadvantages of the Existing Prediction Systems**

1. Limited Attributes. Features like Psychological and Social factors were not considered. These are highly influencing factors that affect student academic outcome.
2. Only one Classifier was used in each system. More Classifiers should have been used to provide a robust results
3. The number of data set was limited.
4. Data collected was only for one year. It should be expanded by adding data from different years
5. Using Naïve Bayes for prediction takes more runtime memory
6. Accuracy can be severely degraded by the presence of noisy or irrelevant features in KNN

### **3.1.2. Analysis of the new System**

Nowadays, e-education and e-learning is highly influenced. Everything is shifting from manual to automated systems. In this research, a system is developed to predict Postgraduate Student academic performance by analysing the students' performance in their first semester courses using K-Bay Classification Method; an integration of K-nearest neighbor classifier and Naïve Bayes. Attributes of the student such as demographic factors, social and academic related factors, unit-test marks, last semester exams and aggregate Cumulative Grade Point Average (CGPA) of the student in the previous semesters will be used in the Prediction.

Combining classifiers to improve the accuracy is a common phenomenon now-a-days, being simpler yet powerful algorithms both Naïve Bayes and KNN are ideal candidate for combination to achieve higher accuracy. The hybrid system will be used to compare the two single classifiers- Naïve Bayes and KNN to find the more efficient data mining classifier. This would result to finding out a more efficient and time saving algorithm to predict the performance of a student. The new system will be cost and time efficient. This will have simple operations. By using the model, students, teachers, and curriculum assessors can easily access an up-to-date curriculum of their various Departments.

The Model is developed to assist lecturers in consulting with students by giving lecturers the permission to view the students' past performance. In the event that the advice generated for the student is insufficient, or in a scenario special case that requires human attention; the system allows the student to remotely interact with an advisor associated with the student's course of study via email. At this point, it is left to the discretion of the student and human advisor, as to whether the issue can be handled remotely, or if a face to face appointment needs to be made.

The idea of the new algorithm is very simple. KNN and Naïve Bayes are used to in the training phase and test phase. The two classifiers are then combined to give a powerful classifier that increased the accuracy of the prediction at a lesser time. The new system was also able to predict student's academic performance and offer academic advising based on the academic standing. The new system was able to query the factors that affect each students performance and the important factors that generally affect students academic performance.

The process model of the new algorithm is represented in figure 3.6 below. Beginning with the collection of data from exams units and questionnaire and then followed by pre-processing where the data is transformed into the format that the algorithm can mine. The classification process is done and the result for the analysis is evaluated.

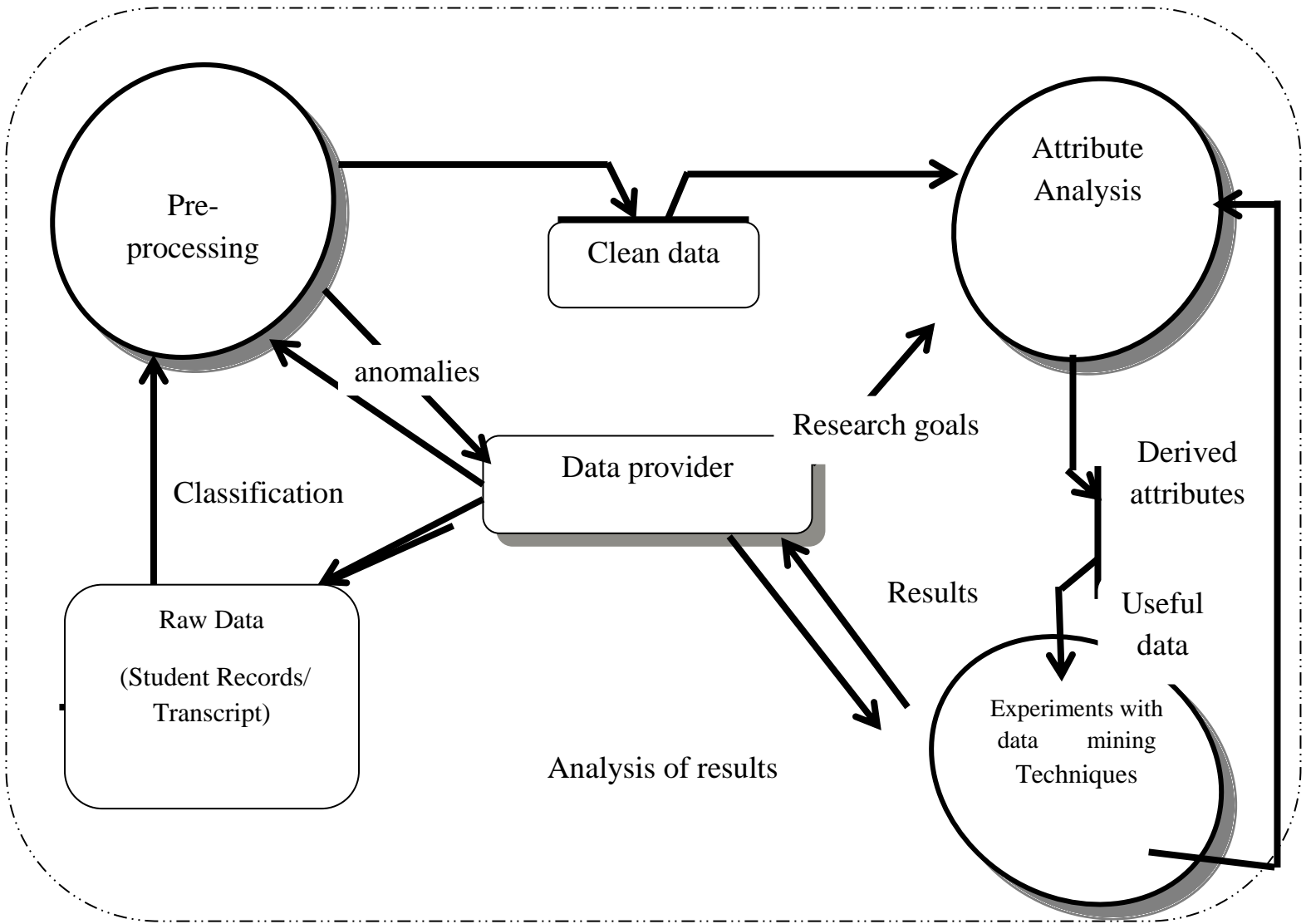


Fig 3.6: Process Model for a Machine Learning Application (Garner et al 1999)

### **3.1.2.1 Data Flow Diagram of the New System**

DFD graphically representing the functions, or processes, which capture, manipulate, store, and distribute data between a system and its environment and between components of a system

The data flow diagram interprets the algorithms' output and gives advice on further experiments that could be run with the student data. The data flow process is illustrated in Figure 3.7 below. The user logs in based on Access Status (Student/ Lecturer/Advisor). On successful log in, for the student, he/she is able to register his or her courses, fill other information that may be required based on the attributes. The student can then go ahead to see the performance prediction. The system predicts the student's performance and further takes a suitable action to improve the student performance by advising the students and proffering solutions based on the student's academic standing. The technique will also monitor and evaluate the student academic performance at different year levels before the final semester in order to predict the grade of the students. The academic adviser can also log in to view students' performance and also generate report on the academic standing of all students in his purview.

The admin updates curriculum and updates other information. He/She also creates accounts for each user. The admin can query the database for each student's information, request for the students academic standing and generate reports on the academic standing of students.

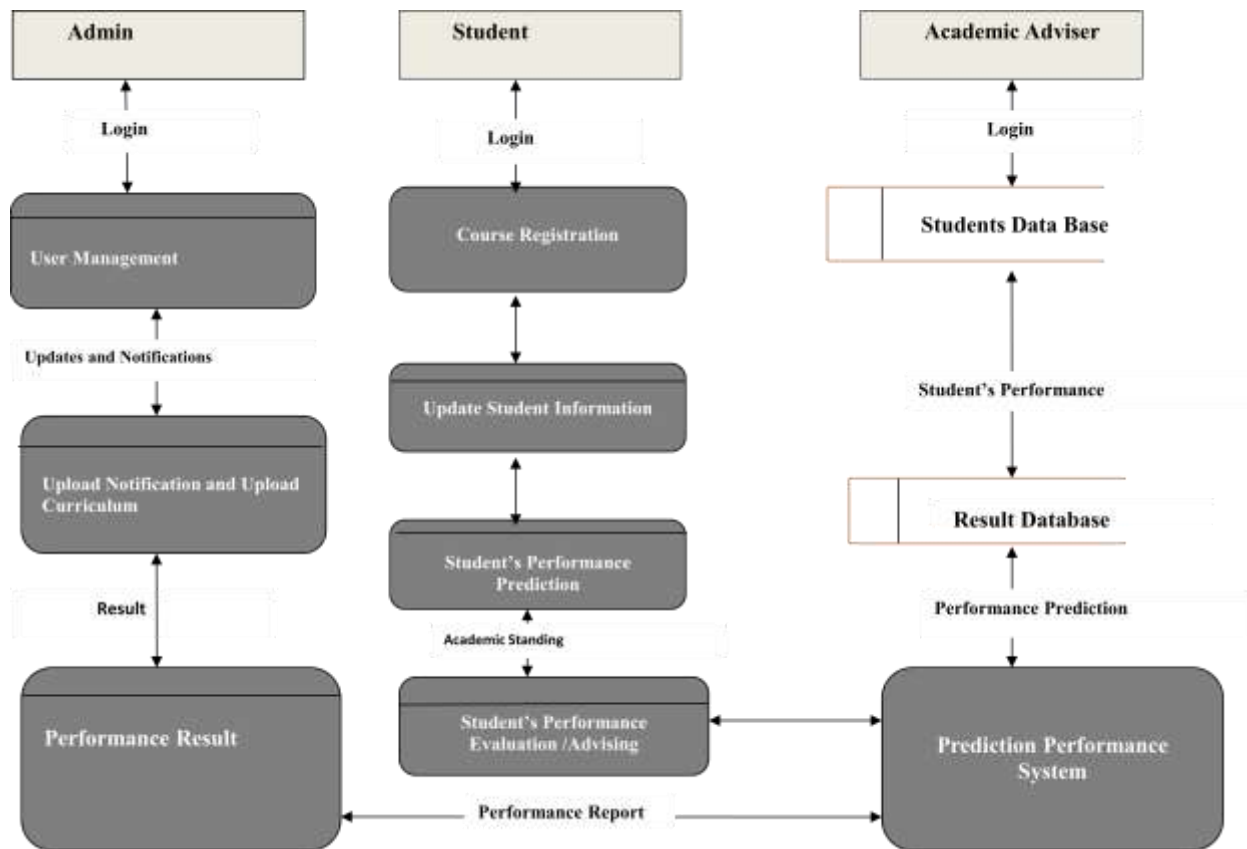


Figure 3.7: Data Flow Diagram of the new System

### 3.1.3 Advantages of the new System.

- 1) Hybridizing the two algorithms will increase the efficiency and performance when compared to individual classifiers. The prediction percentage will also be improved.
- 2) Demographic, Academic Related Factors and Social factors in addition, results from first semester courses will be considered. This will help to produce an efficient performance prediction model
- 3) The new model not only predicts Students score, it identifies factors that affect students' performance. It will help identify weak students who are likely to perform poorly in their studies. This would help in advising students on their areas of weakness and strength.
- 4) Good security is provided by the new system. Security measures are taken to avoid mishandling of database.

- 5) Classification time is reduced in the new system
- 6) The new system provides an efficient data base and search engine. On request, student's information, results and attributes will be displayed.
- 7) The new system will identify and rank influencing factors that contribute to the prediction of students' academic performance. It will also show the ranking of courses that has significant impact on predicting the students' overall academic results
- 8) Large Data set can be trained and tested using the Hybrid of the two models to predict Students' performance

#### **3.1.4 Justification of the New System**

The new system will help to solve the problems inherent in the existing system by providing efficient system that uses educational data mining and multi- agents for the Performance Prediction.

- a. The model will maintain the database in which students transcripts and information are saved.
- b. The model has a higher accuracy in prediction of students performance since it will be the combination of hybrid model that eliminates the problems inherent in each classifier.
- c. The security will be maintained and student's data secured from unauthorised users.
- d. The speed and accuracy of the performance prediction is high to compare with existing model

Once Students Performance Prediction is enhanced in the academic institutions, factors affecting academic performance will be identified, students who are at risk of failing will also be identified and measures will be taken early enough, which will in turn assist academic stakeholders to improve academic performance which is the main goal of study. This justifies the need for the new system

#### **3.2 Methodology Adopted**

Research methodology is a systematic programming approach of a well-defined procedure that should be followed in carrying out a thorough research project.

An adequately suitable methodology that would ensure a very detailed research work and ensured a high degree of accuracy and efficiency will be adopted. The research methodology to



be used will help to ensure that a thorough study of the present system is effectively carried out. This will help project research team to completely understand the existing system, how the new system should be structured and the functionalities needed to address the seemingly problem discovered. Furthermore, it will help to know if there should be a total overhauling of the existing system or if only improvements should be made.

Hence, after due consideration of the above reasons, and Object Oriented Analysis and Design Methodology (OOADM) and Knowledge Discovery in Database (KDD) were adopted in this research.

1. Knowledge Discovery in Databases (KDD) refers to the overall process of discovering useful knowledge from data. It involves the evaluation and possibly interpretation of the patterns to make the decision of what qualifies as knowledge. It also includes the choice of encoding schemes, preprocessing, sampling, and projections of the data prior to the data mining step. The unifying goal of the KDD process is to extract knowledge from data in the context of large databases. It does this by using data mining methods(algorithms) to extract (identify) what is deemed knowledge, according to the specifications of measures and thresholds, using a database along with any required preprocessing, sub- sampling, and transformations of that database.

Data Mining is widely used in educational field to find new hidden patterns from student's data. The hidden patterns that are discovered can be used to understand the problem in the educational field. Data mining techniques provide deep insight into the student's database and provide a suitable needed knowledge for making the decision on the education system by applying some data mining techniques on the students data, some most interesting facts about student's behaviour, student's interest in the study, student's interest in sports etc may come out and further according to these information students may be guided for improving their performance..

The steps involved in the knowledge discovery process include–

- i. Data Cleaning: Removal of noisy and irrelevant data from collection.
- ii. Data Integration: Data integration is defined as heterogeneous data from multiple sources combined in a common source (Data Warehouse).
- iii. Data Selection: Data relevant to the analysis is decided and retrieved from the data collection.

- iv. Data Transformation: Transforming data into appropriate form required by mining procedure.
- v. Data Mining: Data mining is defined as clever techniques that are applied to extract patterns potentially useful.
- vi. Pattern Evaluation: Identifying strictly increasing patterns representing knowledge based on given measures.
- vii. Knowledge representation: Technique which utilizes visualization tools to represent data mining results. It includes generating reports, tables, discriminant rules, classification rules, characterization rules,

2. Object-oriented analysis and design methodology (OOADM) which is adopted in this research work is a set of standards for analysis and development of the students' academic performance prediction system. It uses a formal methodical approach to the analysis and design of information system. Object-oriented design (OOD) elaborates the analysis models to produce implementation specifications. The main difference between object-oriented analysis and other forms of analysis is that the object-oriented approach organizes requirements around objects, which integrate both behaviors (processes) and states (data) modeled after real world objects which the system interacts with. In other traditional analysis methodologies, the two aspects: processes and data are considered separately. The primary tasks in object-oriented analysis (OOA) are:

- a. Find the objects and organize them
- b. Describe how the objects interact
- c. Define the behavior of the objects
- d. Define the internals of the objects

Common models used in OOA are use cases and object models. Use case diagram is a graphic depiction of the interactions among the elements of a system.. Object models describe the names, class relations (e.g. Circle is a subclass of Shape), operations, and properties of the main objects.

### **3.2.1 Sources of Data / Methods of Data Collection**

In order to carry out a detailed analysis of the existing system, both primary and secondary data were collected from different sources. Both secondary and primary data were used to get facts

on the subject. Primary data was collected from the institution and secondary data was collected from literature review that includes understanding and observing available Academic Performance Prediction System. Secondary data was also gathered from a number of sources in order to carry out an insightful investigation into the existing systems, its working procedures, and its mode of operation. Sources include internet sources, journals, books, newspapers and manual auditing of academic performance prediction.

### **3.2.2 Data Collection and Preparation**

The data set used in this study was obtained from the Department of Accountancy Nnamdi Azikiwe University, Awka Anambra State. A sample containing approximately 1000 postgraduate students were first collected from various Faculties. During this phase, data collection process was studied and in order to select appropriate data set to work with. At this stage, the Department of Accountancy was chosen because of its large number of postgraduate students and also because the statistics of drop outs in the programme was high. The rules and procedures for collecting data about examination results were also reviewed. Records of 499 Masters Students were taken from the records. Their first semester result from 2014 to 2017 academic year was also used. A total of nine courses are selected for student and recorded. Students' demographic data, behavior and attitude data, parent and school factors were collected using questionnaires that are given to the students. These questionnaires were uploaded on google form and the links were sent to the students to enable them fill the questionnaire. Hard copies were also distributed to students. First semester results were collected from the Exams Unit of the Postgraduate School. Courses include both Core and Elective Courses. Result showed the candidates overall performance in each course (totaling 100). The Cumulative Grade Point Average (CGPA) for the first semester for each student was calculated.

This step was followed by data preprocessing step, which is concerned with transforming the collected data into a suitable format in order to be used for performing structured analytics. After that, the research work use discretization mechanism to transform the students' related factors and student grade performance grade from numerical values into nominal values, which represents the class labels of the classification problem. To accomplish this step, the research

work divided the data set into three nominal intervals (High Level, Medium Level and Low Level) based on student's CGPA.

In this research "hybrid" method (combination of Naïve bayes and KNN algorithm) was applied to provide an accurate evaluation for the features that may have an impact on the performance/grade level of the students, and to improve the performance of student's prediction model. These methods resample the original data into samples of data set, and then each sample will be trained by a classifier. The classifiers used in student's prediction model were K-nearest neighbour (KNN) and Naïve Bayesian (NB). Individual classifiers results were then combined through a voting process; the class chosen by most number of classifiers.

In order to choose a tool and a best algorithm to serve as a base in developing the new model for multi agent student academic performance prediction, WEKA, JADE and Netbeans IDE were chosen. The academic results of the previous semester based on 48 attributes like Student registration number, semester results etc contributing a major part in semester internal marks were used to predict the final exam results. Table 3.1 contains attributes as included in the questionnaire and their scaling factors. The attributes were analysed on a 1-5 scale basis while the course and grade point average were scaled based on the University approved result format.

Table 3.1 : Attributes from Questionnaire and Scaling factors

s/n	Attribute	Total Scaling Factors (1,2,3,4,5)
<b>A</b>	<b>DEMOGRAPHIC FACTORS</b>	
1	Registration Number	
2	Sex	Female, Male
3	Marital Status	Single, Married, Divorced
4	Program	PGD, Masters, PhD
5	City of Residence	Rural, Urban
6	Mode of Study	Full Time, part Time
7	Students' Age	20-30, 31-40, 41-50, 51 and above
8	Employment Status	Full Time, part Time, Unemployed
9	Family size	3, 4, 5 and above
10	Job-Course-Relationship	Closely related, Somewhat related, Not related, Unemployed
11	Sponsor	Self -Sponsor, Parents/Guardian, Organization/Scholarship, TETFUND

12	Sponsor's Qualification	Primary School, Junior Secondary, Senior Secondary, Higher Institution
13	Program Motivation	Better Employment Prospect To upgrade qualification Pursuit of Knowledge Personal Fulfilment Lack of Employment
<b>B</b>	<b>ACADEMIC/WORK RELATED FACTOR</b>	
14	Non-adherence to Academic calendars by Lecturers	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
15	Students not being present for Lectures and Other postgraduate activities	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
17	Supervisor not specialized in student's area of Research	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
18	Supervisor is too busy with extensive commitment	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
19	Incompatibility with supervisor	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
20	Supervisor is not up to date in the field	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
21	Lack of Commitment from the Supervisor	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
22	Supervisor not always available to devote sufficient time for supervision	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
23	Supervisor's lack of expertise on students topic	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
24	Modality of study conflicts with Student's employment	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
25	Lack of access to research materials	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
26	Difficulties in generating researchable topic	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
27	Lack of ICT knowledge of research method	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
28	Insufficient knowledge of research method	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
29	Frequent closure due to strike actions	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
30	Lack of proper guidance	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)

31	Problem of Funding	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
32	Problem of accommodation	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
33	Poor library facilities, Standard equipment and Laboratory	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
34	Untimely submission of Postgraduate Semester results	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
<b>C</b>	<b>SOCIAL FACTOR</b>	
35	Keeping large numbers of friends/family	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
36	Regular hangout with friends/Family regularly	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
37	Use of stimulants/drugs enhance study	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
38	Regular use of the internet for surfing and social networking	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
39	Encouragement from Sponsors/ Partner in Postgraduate Pursuit	Strongly Disagree(1), Disagree (2), Agree (3), Strongly Agree (4), Indifferent (5)
40	Courses	ACC 811, ACC 813, ACC 815, ACC 817, ACC 819, ACC 821, ACC 825, ACC 827, ACC 829
41	Grades	A- 70% -100%, B- 60-69%, C- 50-59% D- 45-49%, E- 40-44%, F- 0-39%
42	Grade Point Average	A- 5.00, B-4.00, C- 3.00, D- 2.00, E- 1.00, F- 0

The analysis on the various attributes/factors as gotten from the questionnaires and their frequency and percentage are shown in table 3.2, 3.3 and 3.4. The total number of instances used was 499m instances.

Table 3.2: Analysis of Demographic Factors

	Attribute	Scaling Factors	Frequency	Percentage
<b>DEMOGRAPHIC FACTORS</b>				
1	Registration Number			
2	Sex	Female	230	46.09218
		Male	269	53.90782
3	Marital Status	Single	162	32.46493
		Married	337	67.53507
		Divorced	0	0
		Masters	499	100
5	City of Residence	Rural	40	8.016032
		Urban	459	91.98397
6	Mode of Study	Full Time	312	62.52505
		Part Time	187	37.47495
7	Students' Age	20-30	130	26.0521
		31-40	258	51.70341
		41-50	94	18.83768
		51 and above	17	3.406814
8	Employment Status	Full Time	341	68.33667
		Part Time	87	17.43487
		Unemployed	71	14.22846
9	Family size	3 and less	63	12.62525
		4 sizes	156	31.26253
		5 and above	280	56.11222
10	Job-Course-Relationship	Closely related	231	46.29259
		Somewhat related	92	18.43687
		Not related	116	23.24649
		Unemployed	60	12.02405
11	Sponsor	Self Sponsor	403	80.76152
		Parents/Guardian	73	14.62926
		Organization/Scholarship	16	3.206413
		TETFUND	7	1.402806
12	Sponsor's Qualification	Primary School	156	31.26253
		Junior Secondary	20	4.008016
		Senior Secondary	115	23.04609
		Higher Degree	208	41.68337
13	Program Motivation	Better Employment Prospect	154	30.86172
		To upgrade qualification	209	41.88377
		Pursuit of Knowledge	38	7.61523
		Personal Fulfilment	51	10.22044
		Lack of Employment	47	9.418838

Table 3.3: Analysis of Score

	<b>Exam GRADE</b>	<b>Scaling Factor</b>	<b>Frequency</b>	<b>Percentage</b>
1	ACC 811 (Core)	<b>A</b>	95	19.03808
		<b>B</b>	163	32.66533
		<b>C</b>	169	33.86774
		<b>D</b>	50	10.02004
		<b>E</b>	8	1.603206
		<b>F</b>	14	2.805611
2	ACC 813 (Core)	<b>A</b>	96	19.23848
		<b>B</b>	166	33.26653
		<b>C</b>	165	33.06613
		<b>D</b>	48	9.619238
		<b>E</b>	13	2.60521
		<b>F</b>	14	2.805611
3	ACC 815 (Core)	<b>A</b>	54	10.82164
		<b>B</b>	165	33.06613
		<b>C</b>	201	40.28056
		<b>D</b>	57	11.42285
		<b>E</b>	13	2.60521
		<b>F</b>	9	1.803607
4	ACC 817 (Core)	<b>A</b>	60	12.02405
		<b>B</b>	169	33.86774
		<b>C</b>	186	37.27455
		<b>D</b>	63	12.62525
		<b>E</b>	15	3.006012
		<b>F</b>	6	1.202405
5	ACC (Core)	<b>A</b>	80	16.03206
		<b>B</b>	160	32.06413
		<b>C</b>	177	35.47094
		<b>D</b>	62	12.42485
		<b>E</b>	16	3.206413
		<b>F</b>	4	0.801603
4	ACC 821 (Core)	<b>A</b>	108	21.64329
		<b>B</b>	151	30.26052
		<b>C</b>	141	28.25651
		<b>D</b>	72	14.42886
		<b>E</b>	12	2.40481
		<b>F</b>	15	3.006012
45	ACC 825/827/829 (Elective)	<b>A</b>	102	20.44088
		<b>B</b>	143	28.65731
		<b>C</b>	159	31.86373
		<b>D</b>	69	13.82766
		<b>E</b>	11	2.204409
		<b>F</b>	15	3.006012



Table 3.4: Analysis of Demographic Factors

<b>B</b>	<b>ACADEMIC/WORK RELATED FACTOR</b>	<b>Scaling Factor</b>	<b>Frequency</b>	<b>Percentage</b>
14	Non-adherence to Academic calendars by Lecturers	STRONGY DISAGREE	13	2.60521
		DISAGREE	48	9.619238
		AGREE	288	57.71543
		STRONGLY AGREE	121	24.2485
		INDIFFERENT	29	5.811623
15	Students not being present for Lectures and Other postgraduate activities	STRONGY DISAGREE	92	18.43687
		DISAGREE	124	24.8497
		AGREE	182	36.47295
		STRONGLY AGREE	31	6.212425
		INDIFFERENT	70	14.02806
16	Supervisor not specialized in student's area of Research	STRONGY DISAGREE	91	18.23647
		DISAGREE	112	22.44489
		AGREE	188	37.67535
		STRONGLY AGREE	86	17.23447
		INDIFFERENT	22	4.408818
17	Supervisor is too busy with extensive commitment	STRONGY DISAGREE	84	16.83367
		DISAGREE	224	44.88978
		AGREE	120	24.0481
		STRONGLY AGREE	59	11.82365
		INDIFFERENT	12	2.40481
18	Incompatibility with supervisor	STRONGY DISAGREE	170	34.06814
		DISAGREE	128	25.6513
		AGREE	77	15.43086
		STRONGLY AGREE	109	21.84369
		INDIFFERENT	15	3.006012
19	Supervisor is not up to date in the field	STRONGY DISAGREE	30	6.012024
		DISAGREE	32	6.412826
		AGREE	282	56.51303
		STRONGLY AGREE	141	28.25651
		INDIFFERENT	14	2.805611
20	Lack of Commitment from the Supervisor	STRONGY DISAGREE	19	3.807615
		DISAGREE	59	11.82365
		AGREE	350	70.14028
		STRONGLY AGREE	63	12.62525
		INDIFFERENT	8	1.603206
21	Supervisor not always available to devote sufficient time for supervision	STRONGY DISAGREE	51	10.22044
		DISAGREE	113	22.64529
		AGREE	193	38.67735
		STRONGLY AGREE	130	26.0521
		INDIFFERENT	12	2.40481
22	Supervisor's lack of expertise on students topic	STRONGY DISAGREE	8	1.603206
		DISAGREE	60	12.02405
		AGREE	72	14.42886

		<b>STRONGLY AGREE</b>	353	70.74148
		<b>INDIFFERENT</b>	6	1.202405
23	Modality of study conflicts with Student's employment	<b>STRONGY DISAGREE</b>	3	0.601202
		<b>DISAGREE</b>	24	4.809619
		<b>AGREE</b>	236	47.29459
		<b>STRONGLY AGREE</b>	138	27.65531
		<b>INDIFFERENT</b>	92	18.43687
24	Lack of access to research materials	<b>STRONGY DISAGREE</b>	150	30.06012
		<b>DISAGREE</b>	178	35.67134
		<b>AGREE</b>	112	22.44489
		<b>STRONGLY AGREE</b>	46	9.218437
		<b>INDIFFERENT</b>	13	2.60521
25	Difficulties in generating researchable topic	<b>STRONGY DISAGREE</b>	89	17.83567
		<b>DISAGREE</b>	50	10.02004
		<b>AGREE</b>	298	59.71944
		<b>STRONGLY AGREE</b>	57	11.42285
		<b>INDIFFERENT</b>	5	1.002004
26	Lack of ICT knowledge of research method	<b>STRONGY DISAGREE</b>	23	4.609218
		<b>DISAGREE</b>	183	36.67335
		<b>AGREE</b>	218	43.68737
		<b>STRONGLY AGREE</b>	67	13.42685
		<b>INDIFFERENT</b>	8	1.603206
27	Insufficient knowledge of research method	<b>STRONGY DISAGREE</b>	12	2.40481
		<b>DISAGREE</b>	44	8.817635
		<b>AGREE</b>	281	56.31263
		<b>STRONGLY AGREE</b>	150	30.06012
		<b>INDIFFERENT</b>	11	2.204409
28	Frequent closure due to strike actions	<b>STRONGY DISAGREE</b>	21	4.208417
		<b>DISAGREE</b>	26	5.210421
		<b>AGREE</b>	315	63.12625
		<b>STRONGLY AGREE</b>	127	25.4509
		<b>INDIFFERENT</b>	10	2.004008
29	Lack of proper guidance	<b>STRONGY DISAGREE</b>	20	4.008016
		<b>DISAGREE</b>	28	5.611222
		<b>AGREE</b>	284	56.91383
		<b>STRONGLY AGREE</b>	111	22.24449
		<b>INDIFFERENT</b>	56	11.22244
30	Problem of Funding	<b>STRONGY DISAGREE</b>	18	3.607214
		<b>DISAGREE</b>	40	8.016032
		<b>AGREE</b>	168	33.66733
		<b>STRONGLY AGREE</b>	261	52.30461
		<b>INDIFFERENT</b>	12	2.40481
31	Problem of accommodation	<b>STRONGY DISAGREE</b>	28	5.611222
		<b>DISAGREE</b>	53	10.62124
		<b>AGREE</b>	364	72.94589

		<b>STRONGLY AGREE</b>	37	7.41483
		<b>INDIFFERENT</b>	17	3.406814
32	Poor library facilities, Standard equipment and Laboratory	<b>STRONGY DISAGREE</b>	9	1.803607
		<b>DISAGREE</b>	20	4.008016
		<b>AGREE</b>	322	64.52906
		<b>STRONGLY AGREE</b>	133	26.65331
		<b>INDIFFERENT</b>	15	3.006012
33	Untimely submission of Postgraduate Semester results	<b>STRONGY DISAGREE</b>	59	11.82365
		<b>DISAGREE</b>	49	9.819639
		<b>AGREE</b>	248	49.6994
		<b>STRONGLY AGREE</b>	118	23.64729
		<b>INDIFFERENT</b>	25	5.01002
34	Keeping large numbers of friends/family	<b>STRONGY DISAGREE</b>	15	3.006012
		<b>DISAGREE</b>	275	55.11022
		<b>AGREE</b>	129	25.8517
		<b>STRONGLY AGREE</b>	70	14.02806
		<b>INDIFFERENT</b>	10	2.004008
35	Regular hangout with friends/Family regularly	<b>STRONGY DISAGREE</b>	21	4.208417
		<b>DISAGREE</b>	115	23.04609
		<b>AGREE</b>	295	59.11824
		<b>STRONGLY AGREE</b>	49	9.819639
		<b>INDIFFERENT</b>	19	3.807615
36	Use of stimulants/drugs enhance study	<b>STRONGY DISAGREE</b>	277	55.51102
		<b>DISAGREE</b>	183	36.67335
		<b>AGREE</b>	21	4.208417
		<b>STRONGLY AGREE</b>	16	3.206413
		<b>INDIFFERENT</b>	2	0.400802
37	Regular use of the internet for surfing and social networking	<b>STRONGY DISAGREE</b>	38	7.61523
		<b>DISAGREE</b>	57	11.42285
		<b>AGREE</b>	290	58.11623
		<b>STRONGLY AGREE</b>	88	17.63527
		<b>INDIFFERENT</b>	26	5.210421
38	Encouragement from Sponsors/ Partner in Postgraduate Pursuit	<b>STRONGY DISAGREE</b>	26	5.210421
		<b>DISAGREE</b>	33	6.613226
		<b>AGREE</b>	339	67.93587
		<b>STRONGLY AGREE</b>	83	16.63327
		<b>INDIFFERENT</b>	18	3.607214

Table 3.5 shows the classification table on which student's Grade point and academic standing was predicted. Multi-classification was used in this work. The minimum Final Cumulative Grade Point Average (FCGPA) for a student be admitted for PhD is 3.50, hence the "Medium" classification was pegged at a minimum of 3.50

Table 3.5 Classification of Students based on Academic Standing (GPA)

S/N	GPA Range	Class
1	4.01-5.00	High
2	3.50-4.00	Medium
3	0.01-3.49	Low

#### **3.2.4. High Level Model of the new System**

The high level model of the new system is shown in figure 3.8. From Data Collection and segregating the data according to the requirement, Pre-Processing, Classification using the model, Training the model, testing the model, Hybridizing the model, Prediction of Students Score and E-Advising of the student based on the academic performance. In the new work KNN, Naive Bayes and a Hybrid algorithm of KNN and Naïve Bayes were used for student's performance prediction.

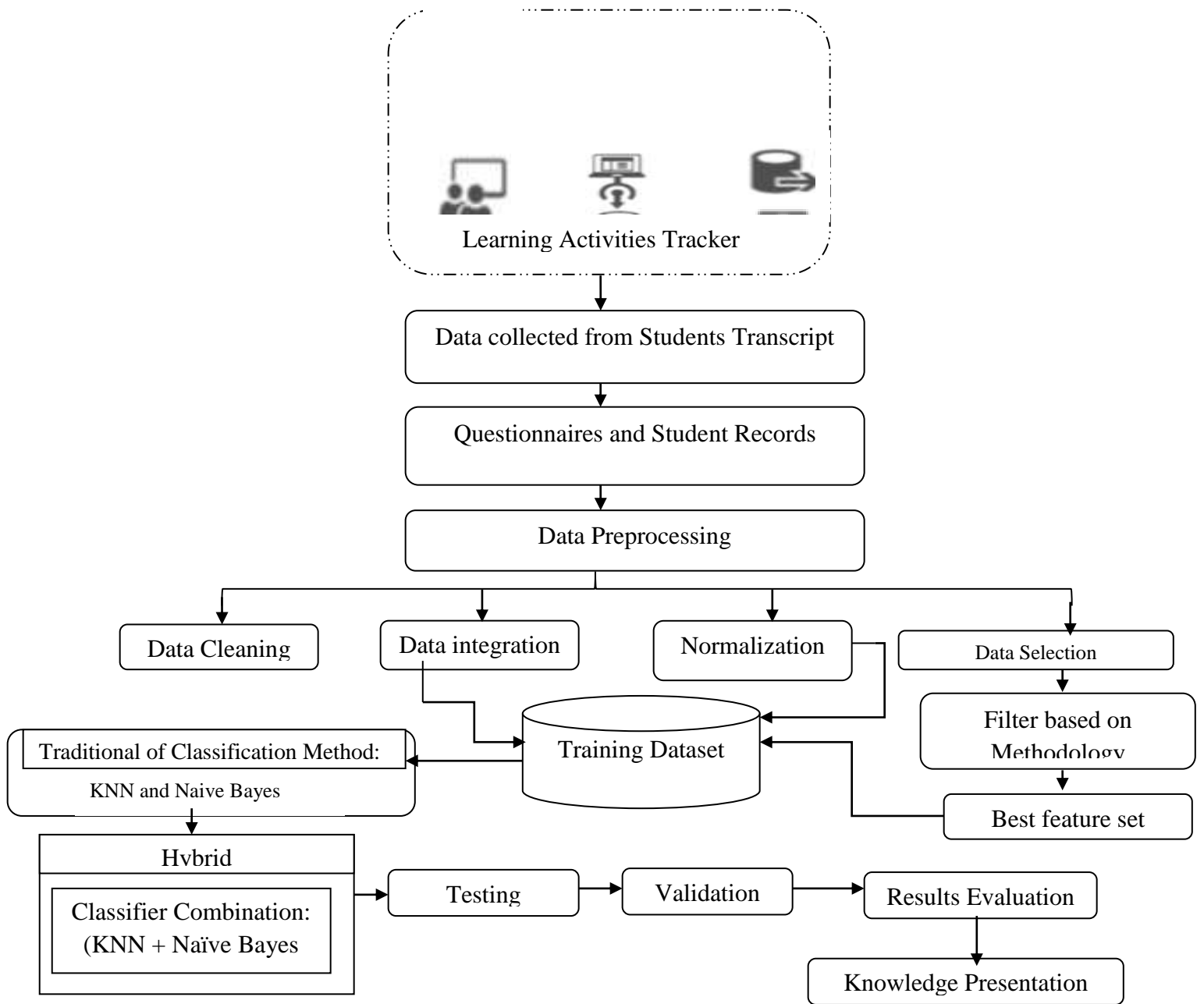


Figure 3.8: High Level Model of the new System

## CHAPTER FOUR

### SYSTEM DESIGN AND IMPLEMENTATION

This chapter provides the details of system design and implementation of student academic performance prediction. The aim is to develop a hybrid prediction model for classifying student academic performance. The model of the new system is shown in figure 4.1. Data was collected from multiple sources: via questionnaire and academic unit of the school. These data was pre-processed to transform the data from the raw form to a much more usable or desired form before feeding in the algorithms. It involved data cleaning (removal of noisy and irrelevant data from the collection), data integration, data normalization (transforming numerical values such as GPA parameters to nominal or categorical class), and data selection (selecting those features which are more informative or relevant). The data was then partitioned into two sets (The training and test data) using 90% and 10% percentage split respectively. Two algorithms, Naïve Bayes and K-Nearest Neighbour (KNN) algorithm were applied to the data. A total of 499 instances were used for the analysis. 10 folds cross validation was used to validate the stability of the model. Results from Naïve Bayes and KNN classification was then combined and used to develop the hybrid model (K-Bay).The results was evaluated, thereafter performance prediction and academic advising were done based on the result.

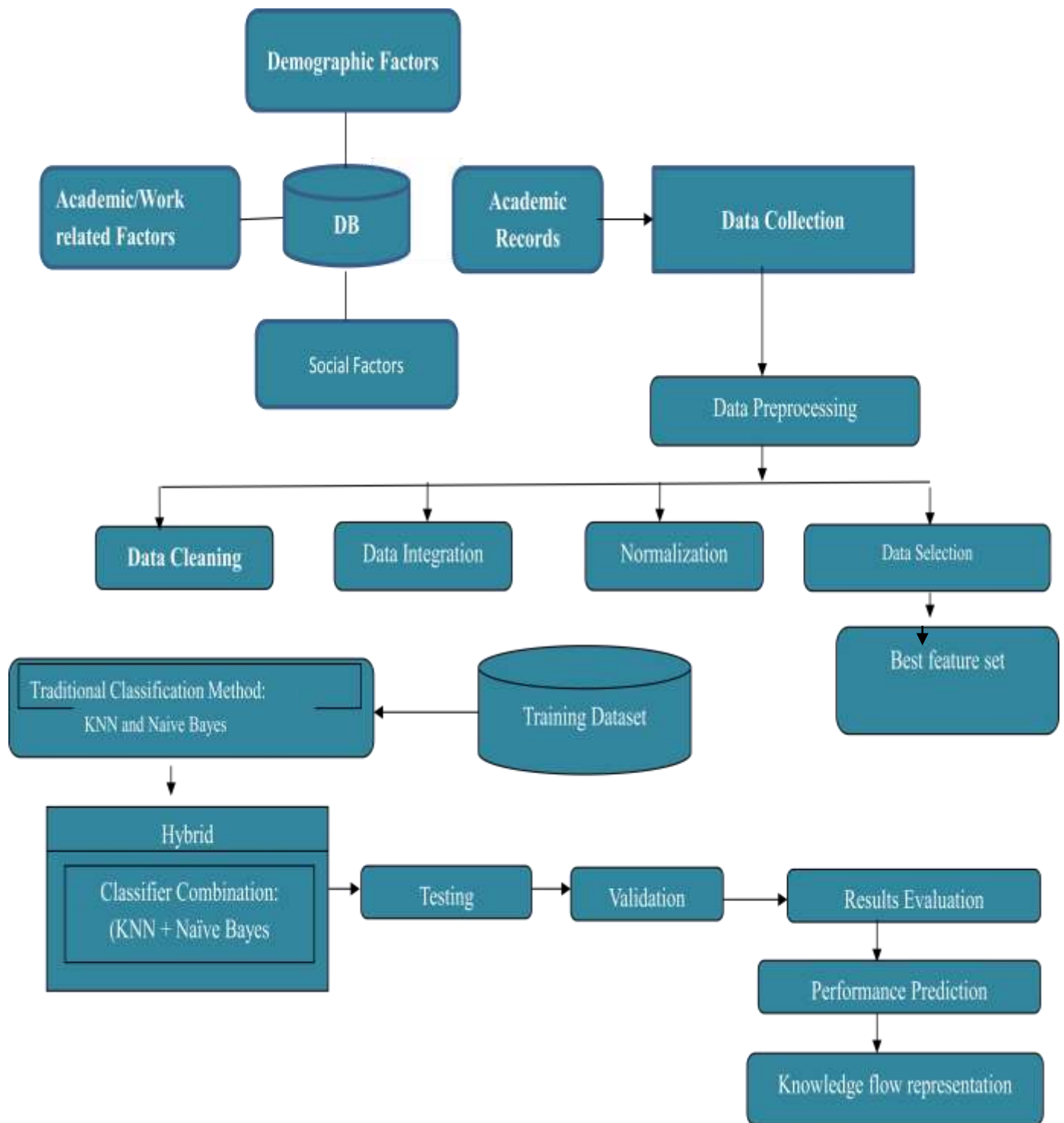


Figure 4.1 Model of the New System

#### **4.1 Objectives of the Design**

- i) Hybridizing two classifiers (KNN and Naves Bayes) to predict academic performance of postgraduate students.
- ii) Categorize the class of student using the obtained grade in first semester
- iii) Identify highly influencing predictive variables on the academic performance of Postgraduate Students
- iv) Identifying and analyzing the main attribute for evaluating students' performance
- v) Predict result and estimate accuracy parameters from the test data set.
- vi) Evaluate the performance of the system.

#### **4.2 Control Centre/Main Menu**

The main menu contains the modules on the Students performance Prediction system. Access to the system is controlled through the user password; which now determines what the user can have access to on the system. Menu in the Control Menu include the File Module, Student Account Registration Module, Student/Admin Account Module, Course Registration Module, Structured dataset Module, Model Building/Cross Validation Module, Training/Testing Module, Hybrid Model Module, Performance Prediction/E-Advisor Module and Search Module as shown in figure 4.2.



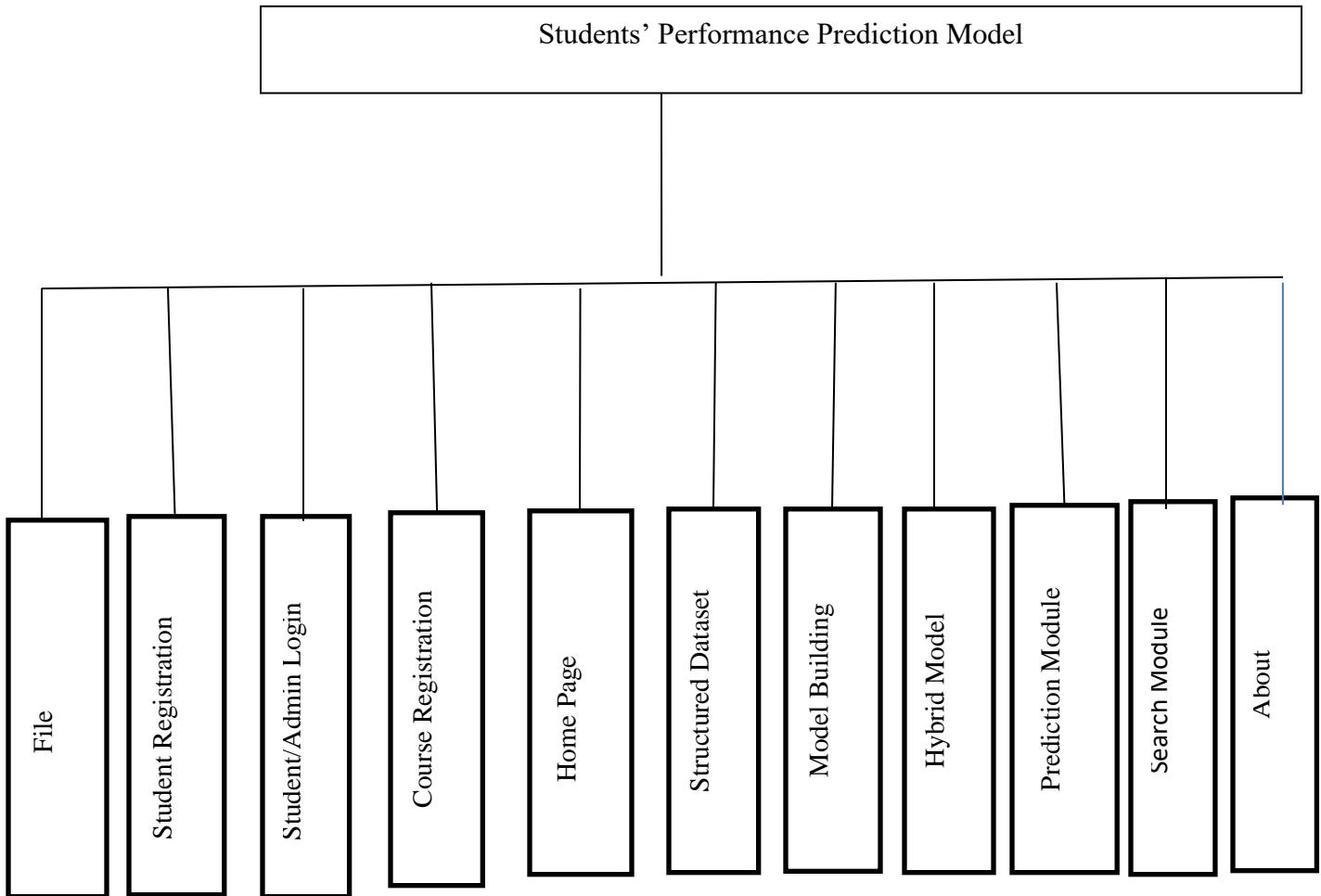


Figure 4.2: Main Menu/Control Menu

### 4.3 The Submenus/Subsystems

The Students Performance prediction System was divided into sub systems. It was designed using Top –Down Approach. The system is structured in a way that each subsystem is accessed from the main menu and executed independently. The sub menus / sub systems are as follows:

#### 4.3.1 Student Account Registration:

This module contains the various stages of student account registration. Creation of account for newly admitted students is done in this module. Admin has to create account for each student before he/she can gain access to their semester course form and perform other task as required.

The module contains student ID, student registration number, session, and semester as shown in figure 4.3

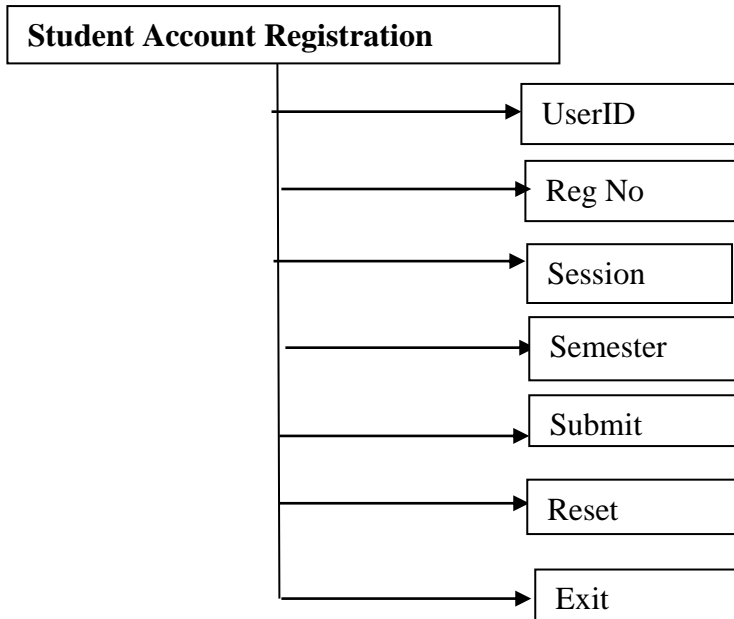


Figure 4.3: Student Account Registration

### 4.3.2 User Account Creation

This module is enables the creation of accounts for admin/teacher/lecturer account. The admin is solely responsible for cresting the account. During account creation, the admin specifies the rolse of the user. This role /position level must be indicated to know the level of access the user will be given. Information hereunder include User ID, User Name, Password, Role, Submit, Reset and Exit as shown in figure 4.4

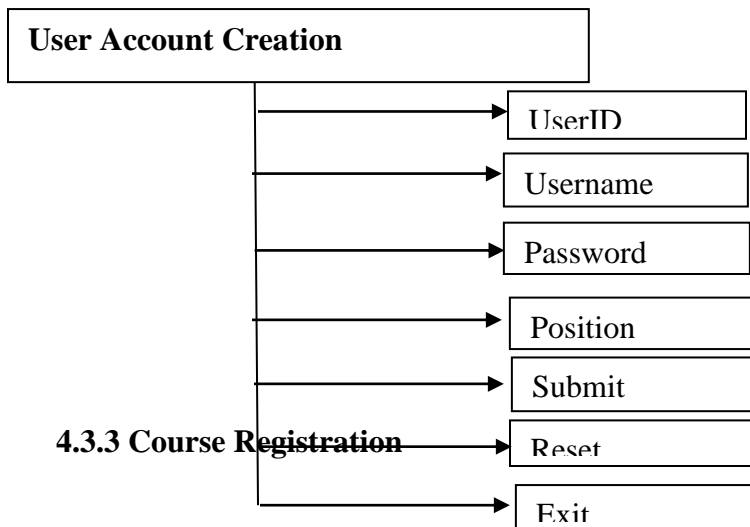


Figure 4.4 User Account Creation

### 4.3.3 Course Registration

In this module, the student is prompted to register his/her courses for the semester. At the beginning of every semester, the student is requested to register their courses. The student has to select all core courses and then chose from the electives depending on the area of specialty. The system also ensures that the student attains the minimum credit for the programme and doesn't exceed the maximum credit units. Information needed in this module include Name, Registration Number, Session, Semester, Select Course Code, Course Tile, Credit Unit, and Total Credit as shown in figure 4.5.

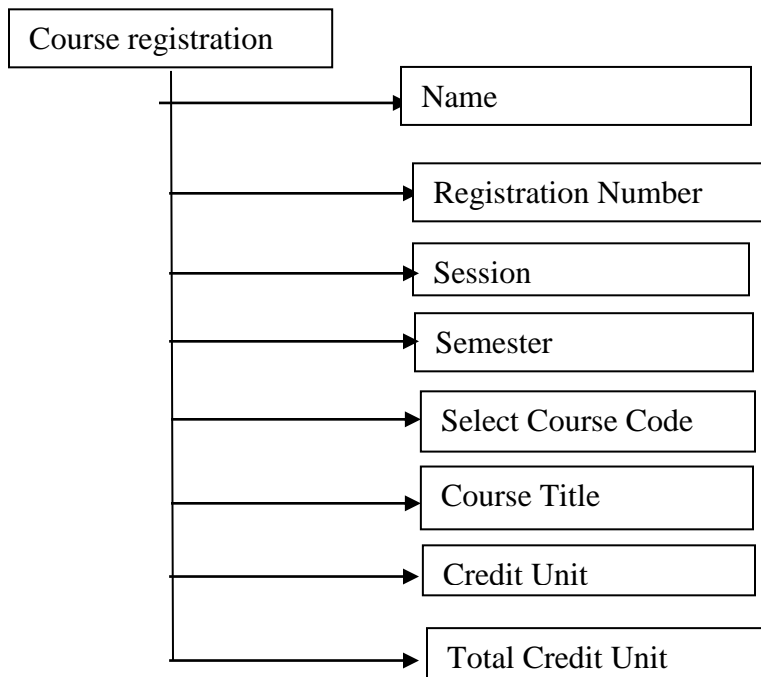


Figure 4.5 Course Registration

#### 4.3.4 Structured Data

This module contains student data used for the prediction. It is segmented into those attributes that affects students' performance which include demographic factors, academic related factors, work related factors, personal factors and students results. The data was preprocessed in excel application package, notepad++, and converted to Comma Separated Value format (CSV) and then Attribute Related File Format (arff) before it was uploaded into MySQL database for further analysis. The Structred Data set consist of the Students results and Attributes as shown in figure 4.6.

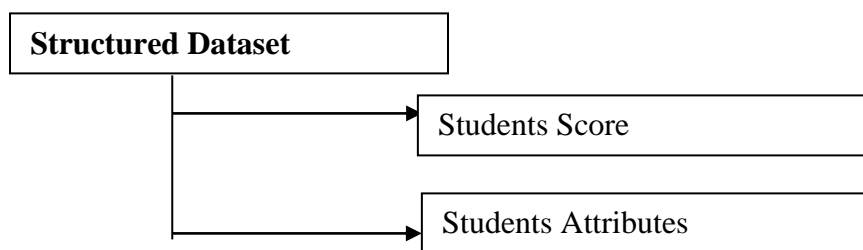


Figure 4.6 Structured Dataset

### 4.3.5 Model Training/Cross-Validation

This module uses preprocessed dataset of student, feature set extraction, classification model and preprocessed data set collected via academic record and questionnaire to implement the model building program module. The model building program module used the element of weka machine learning library, jade library agent and NetBeans IDE to implement the model for full training set. For Cross validation, 10 Fold Cross Validation was used for the analysis. The results of the analysis of the Cross validation was represented in pie chart and bar chart Figure 4.7 shown the diagram of the Model Training.

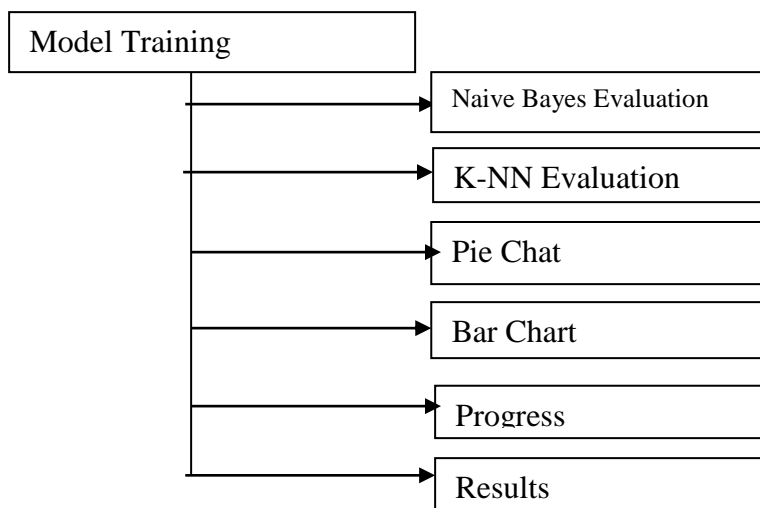


Figure 4.7 Model Training/Cross Validation

### 4.3.6 Training Set/Test Set

Training set is for model building and test is for model evaluation. The training data set is implemented to build the model, while the test (or validation set is used to validate the model.

The entire data was divided using the 90% and 10% split. 90% of the data was used as training set and 10% as test. Classification task, K-NN and Naïve Bayes were used to build the performance model and for performance optimization. Task being performed at each stage

include Naïve Bayes Model Evaluation, K-NN Model Evaluation, Pie Chart, Bar Chart, Progress and Results as shown in Figure 4.8.

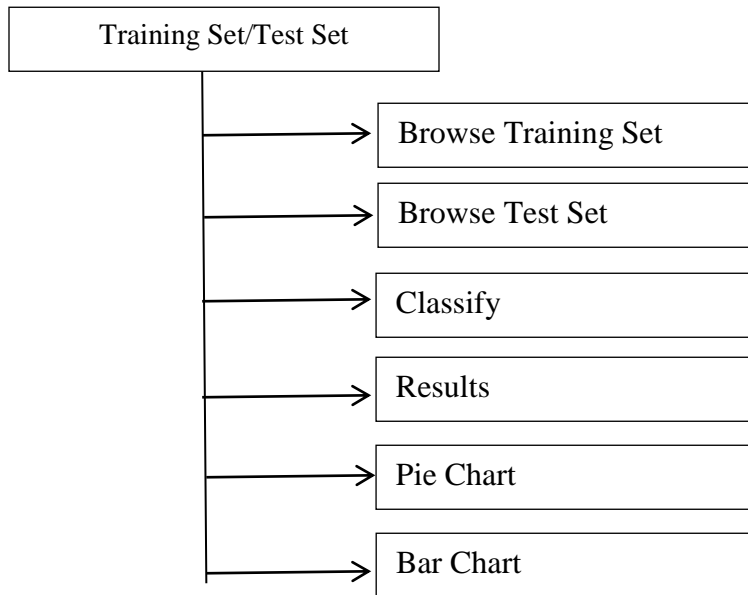
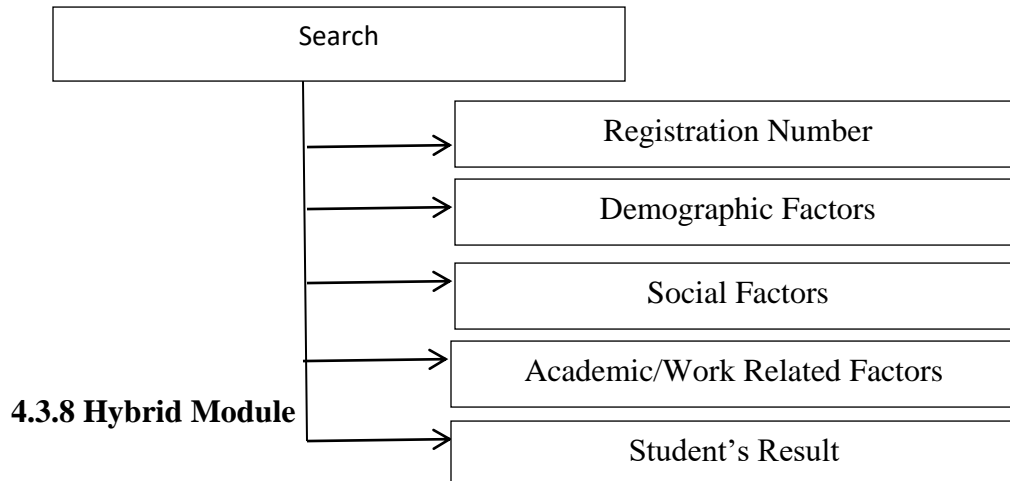


Figure 4.8 Training Set/Test set

#### 4.3.7: Search Menu

The system contains a comprehensive database of the students attributing factors and their academic result. The search Menu when initiated and the student registration number entered, the attributing factors which the students have earlier indicated and academic results are displayed. Task being performed at each stage include Demographic Factors, Social Factors, Academic and Work Related Factors and Student’s Result as shown in Figure 4.9.



#### 4.3.8 Hybrid Module

Figure 4.9: Search Sub Menu

This module shows the content the Hybrid Model. The model combines both KNN and Naïve Bayes technique. It contains the result for both the training and the test set. The result from the test set is used for the hybrid. Figure 4.10 shows the diagram for the hybrid module

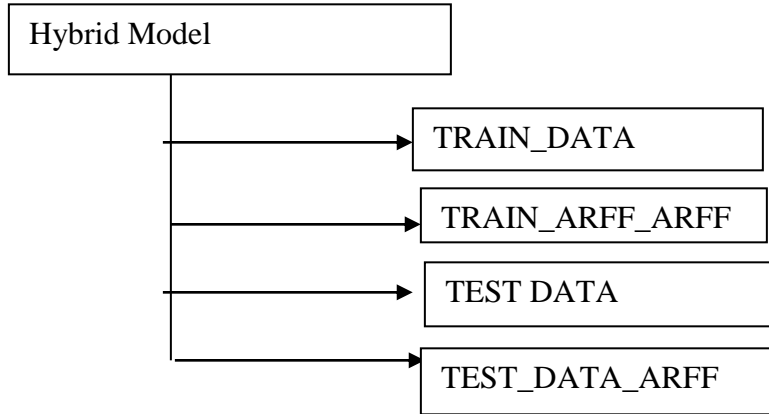


Figure 4.10 Hybrid Model

#### 4.3.9 Performance Prediction/E-Advisor

This model contains the Performance Predictor/E-Advisor. The student selects the model path and enters his/her registration number. The system predicts the grade of the student and advises the student based on his/her performance. Task being perform in each stage such as Model Path, Students Registration number, the Prediction function, Out Put Results, Adviser Module, and Hybrid Result Evaluation as shown in Figure 4.11 below.

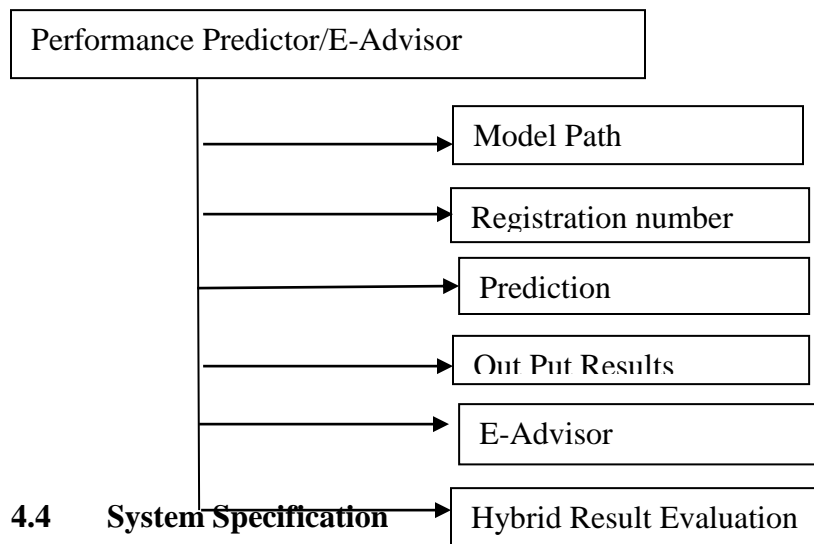


Figure 4.11: Performance Predictor/E-Advisor

The system specification describes the ideal specification for the new system to function optimally. The system specification details are shown in Table 4.1:

Table 4.1 New System Specifications

<b>System specification</b>	<b>Descriptions</b>
Software applications to be used	Java Run Time Environment (JRE) release 8 or later version on the machine that will run the application. WEKA (Waikato Environment for Knowledge Analysis) implements most of the machine learning algorithms and visualizes its results as well. JADE (Java Agent Development Framework) implemented in Java language was also used in the design of the multi agent system. JADE simplifies the implementation of multi-agent systems through a middleware that claims to comply with the FIPA (Foundation for Intelligent Physical Agents) specifications and through a set of tools that supports the debugging and deployment phase. Netbeans IDE, MySQL Database was also used in the design.
Storage requirements	This includes local storage requirements such as hard disk size
System memory	How much RAM will be required by the system in order for it to run effectively
Input devices needed	These include Mouse, Keyboard,
Output devices to be used	These include printers, speakers, visual display unit of 15.6”
Computing/ processing power needed	A standard personal computer with core i3 processor with processing speed of 1.2GHz or higher will be able to access this system
Security and Backup systems	User accounts with rules to be created and maintained by the database administrator
People required	A Database Administrator is needed to update changes to database. It has to be a skilled personnel

The overall purpose of the system specification documentation is to lay down exactly how the system is made up.

#### **4.4.1 Database Development Tool**

Knowledge discovery in databases (KDD) is the process of discovering useful knowledge from a collection of data. This widely used data mining technique is a process that includes data preparation and selection, data cleansing, incorporating prior knowledge on data sets and interpreting accurate solutions from the observed results.

WEKA supports several standard data mining tasks, more specifically, data preprocessing, clustering, classification, regression, visualization, and feature selection. All of WEKA's techniques are predicated on the assumption that the data is available as a single flat file or relation, where each data point is described by a fixed number of attributes (normally, numeric or nominal attributes, but some other attribute types are also supported). WEKA provides access to SQL databases using Java Database Connectivity and can process the result returned by a database query.

#### **4.4.2 Database Design and Structure**

Database names were created and were organized into physical files for speed optimization. The databases were structured into tables, views, rows, size, and columns. Rules were set up to govern the relationships between different data fields. The structure of the tables in the database includes:

- i. Student Dataset Table
- ii. Student Course Form Table
- iii. Student Semester Table
- iv. Admin, Teacher and Student Table
- v. Agent Task Specification

Student's dataset table contains all the variable for the students attributes. It contains the demographic, work related and social factors as well as the students score in each case. The data types and the data sizes of each variable are specified. Table 4.2 describes the student's data structure

Table 4.2: Student Dataset



S/No	Field	Data Type(Size)	Description of Variable
1	StudentRegNo	varchar(23)	It describes student's registration number
2	Mode_Study	int(12)	It describes student's mode of study
3	Gender	int(11)	It describes student's gender
4	Marital_Stat	int(11)	It indicates the marital status
5	Prog	int(13)	It describes the type of program the student is currently running
6	City	int(11)	Student's city of residence
7	Age	int(12)	Student's Age
8	Employ_Stat	int(12)	It describes student's current employment either full-time (35 hours per week or more) or part-time
9	Family-Size	int(14)	Student's Family Size
10	Job_Course	int(13)	It describe how closely related the student's current job is to the skills and concepts of his/her current course.
11	Sponsor	int(14)	Sponsor
12	prog_Motive	int(12)	It describe the reasons for pursuit of postgraduate Studies
13	NAAC	int(13)	Indicates if Non-adherence to Academic calendars by Lecturers is a factor for academic performance
14	Atten_ Lect	int(11)	Indicates if the level of Lecturer's attendance to lecture and Other postgraduate activities is a factor for academic performance
15	Sup_Area	int(13)	Indicates if the Supervisor being specialized in student's area of Research is a factor
16	Sup_Busy	int(14)	Indicates if thethe supervisor being too busy with extensive commitment affects student's performance
17	incompact_Sup	int(11)	Indicates if Incompatibility with supervisor affects student's performance
18	Sup_Not_uptodate	int(11)	Indicates if Supervisor is not up to date in the field affects performance
19	Spu_Commitment	int(10)	Indicates if students performance can be affected by Supervisor's lack of commitment
20	Sup_Unavailablesufficient	int(10)	Indicates if supervisor not being always available to devote sufficient time for supervision is a factor
21	Sup_Expertise	int(10)	Indicates if supervisor lacks expertise on students topic affects performance
22	Study_conflits_job	int(11)	Indicates if the modality of study conflicts with student's employment
23	Access-Internet	int(8)	Indicates if lack of access to research materials affect performance
24	Diff_Research_Topic	int(7)	Indicates if difficulties in generating researchable topic is a factor
25	Lack_ICT Knowledge	int(9)	Indicates the lack of ICT knowledge of research method
26	Insuff_Know_Research	int(6)	Indicates if Insufficient knowledge of research method
27	Strike	int(6)	Indicates if the frequent closure due to strike actions affects performance

28	Lack_Proper Guide	int(8)	Indicates if the Lack of proper guidance affects performance
29	Funding_Prob	int(7)	Indicates if funding is a major problem in student's academic pursuit
30	Accom_Prob	int(8)	Indicates if accommodation is a major problem in student's academic pursuit
31	poor_Lib_Equip_Lab	int(4)	Indicates if poor library facilities, Standard equipment and Laboratory affects performance
32	Sub_Result	int(7)	Indicates if Postgraduate Teachers not submitting Semester results on time affects performance
33	Keep_No_of Friends	int(4)	Describes if keeping considerable number of friends/family affects performance
34	Regular_Hangout	int(6)	Describes if affects performance is affected by regular hang out with friends/Family
35	Use_of_Stimulant	int(7)	Describes if the use of stimulants/drugs enhances student's study
36	Access_Internet	int(12)	Indicates if regular access to social media and internet affect affects performance
37	Sponsor_Partner_Enourage	int(7)	Indicates if encouragement from Parents /Partner in PG Pursuit has effect on academic performance
38	ACC 811	int(8)	Student's Score in ACC811
39	ACC 813	int(9)	Student's Score in ACC813
40	ACC 815	int(7)	Student's Score in ACC815
41	ACC 817	int(8)	Student's Score in ACC817
42	ACC 819	int(12)	Student's Score in ACC819
43	ACC 821	int(9)	Student's Score in ACC821
44	ACC 825	int(23)	Student's Score in ACC825
45	ACC827	int(23)	Student's Score in ACC827
46	ACC829	int(22)	Student's Score in ACC829
47	CGPA	Float	Cumulative grade point of all the scores
48	CLASS	varchar(23)	Class of degree

Students Course registration table contains the variables for the registration of courses, their data size and their data type. It also describes what each field contains. Table 4.3 contains the Students Course Form information.

Table 4.3: Student Course Form Table

S/No	Field	Data Type(Size)	Description of Variable
1	Role	int(11)	It describe student authentication ID
2	FullName	varchar(30)	It describe students' full name
3	RegNo	varchar(23)	It describe student registration number
4	Session	varchar(22)	It indicate the student session

5	Semester	varchar(22)	It describes the student semester
6	CourseCode_*	varchar(22)	Course code for various Courses
14	CourseTitle_*	varchar(50)	Course title for various courses

The User table contains the variables, data sizes and data types for the creation of account for each user. It describes the function of each variables and their field. The User table is displayed in Table 4.4.

Table 4.4: User Table

S/N	Field	Data type (size)	Description
1	StudentID	Int(11)	It describe student Authentication unique identification
2	StudentReg	Varchar(40)	It describe student registration number
3	Year	Varchar(25)	It describe student current of running the program
4	Role	Varchar(21)	It describe student current semester such as first semester, second semester and third semester

Four intelligent agents were deployed in the work; they are the user interface agents, model full training agent, model evaluation agent and performance prediction/advisor agents. These agents make up the multi agents that communicate with each other to perform various tasks. The specific task of each agent and their mode of operation are specified in Table 4.5.

Table 4.5: Agent task specification and their function

S/N	Agent	Specific task	Mode of Operation
1	UserInterfaceAgent	Takes action	Authentication requirement
2	ModelFullTrainingAgent	Builds the model with required library	Jade and Weka
3	ModelEvaluationAgent	Uses the student sample dataset to evaluate the model	Machine learning Classification task
4	PerformancePrediction/Advisor Agent	Classification analysis	Naïve Bayes and K-NN

#### 4.4.3 Maths Specification

##### Naïve Bayes Classifier and K-Nearest Neighbor

In abstract, the probability model for a classifier is a conditional model.

$$P(C | x, \dots, x_n)$$

Over a dependent class variable  $C$  with a small number of outcomes or classes, conditional on several variables  $x$  through  $x_n$ .

Using Bayes' theorem, it may be written as:

$$P(C|x) = \frac{P(C)P(x|C)}{P(x)} \quad (4.1)$$

- ✓  $P(c|x)$ : the posterior probability of class (c, target) given predictor (x, attributes).
- ✓  $P(c)$ : the prior probability of class.
- ✓  $P(x|c)$ : the likelihood, which is the probability of predictor given class.
- ✓  $P(x)$ : the prior probability of predictor.

Which can be simplified as: Posterior =  $\frac{\text{Prior of class} \times \text{Likelihood}}{\text{Evidence}}$

$$x = x_1, x_2, \dots, x_n$$

$$P(C|x_1, \dots, x_n) = \frac{P(C)P(x_1, x_2, \dots, x_n|C)}{P(x_1, x_2, \dots, x_n)} \quad (4.2)$$

Normally, the denominator does not depend on  $C$  and the values of the  $x_1$  which are the features are given, therefore the denominator is always constant which means the interest lies only in the numerator of the fraction.

The numerator is equivalent to the joint probability model  $P(C, x_1, \dots, x_n)$  which can be rewritten as follows, using repeated applications of the definition of conditional probability and so forth.

$$\begin{aligned} &P(C, x_1, x_2, \dots, x_n) \\ &= P(C) P(x_1, \dots, x_n | C) \quad (4.3) \\ &= P(C) P(x_1 | C) P(x_2, \dots, x_n | C, x_1) \\ &= P(C) P(x_1 | C) P(x_2 | C, x_1) P(x_3, \dots, x_n | C, x_1, x_2) \\ &= P(C) P(x_1 | C) P(x_2 | C, x_1) P(x_3 | C, x_1, x_2) P(x_4, \dots, x_n | C, x_1, x_2, x_3) \end{aligned}$$

Now the naive conditional independence assumptions come into play: assume that each attribute  $x_i$  is conditionally independent of every other attribute  $x_j$  for  $j \neq i$ .

This means that  $P(x_i | C, x_j) = P(x_i | C)$  and so the joint model can be expressed as

$$P(C, x_1, \dots, x_n) = P(C)P(x_1 | C)P(x_2 | C)P(x_3 | C)\dots$$

$$= P(C) \prod_{i=1}^n P(x_i | C) \quad (4.4)$$

This means that under the above independence assumptions, the conditional distribution over the class variable  $C$  can be written as

$$P(C | x_1, \dots, x_n) = \frac{1}{Z} P(C) \prod_{i=1}^n P(x_i | C) \quad (4.5)$$

Here,  $Z$  is a scaling factor dependent only on  $x_1, x_2, \dots, x_n$  i.e., a constant if the values of the feature variables are known.

The corresponding classifier is the function *classify* defined as follows:

$$\text{Classify}(x_1, \dots, x_n) = \underset{c}{\operatorname{argmax}} P(C = c) \prod_{i=1}^n P(x_i = x_i | C = c) \quad (4.6)$$

In case of K-Nearest Neighbor, given a training set  $D$  and a test object  $z = (x', y')$ , the algorithm computes the distance (or similarity) between  $z$  and all the training objects  $(x, y) \in D$  to determine its nearest-neighbor list,  $D_z(x)$  is the data of a training object, while  $y$  is its class. Likewise,  $x$  is the data of the test object and  $y'$  is its class).

The Euclidean Distance between two points  $(x_1, y_1)$  and  $(x_2, y_2)$  in the plane is given by the equation

$$\text{Distance } D(x_1, y_1), (x_2, y_2) = \sqrt{\sum_{i=1}^k (x_i - y_i)^2} \quad (4.7)$$

#### 4.4.4 Program Module Specification

The software is structured in such a way that each subsystem is selected and executed independently. The task is divided into several modules, which come together to give the solution to the problem. The modules are as follows:

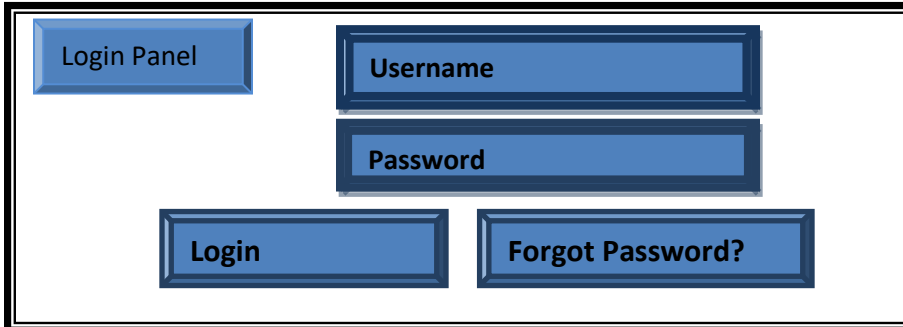
1. **User Account registration:** In this module, the student gives their information to create an account. The information is all about their contact details. They can change their own login and password if the need arises.
2. **Login:** In Login Form module presents students with a form with username and password fields. If the user enters a valid username/password combination they will be granted access to the system.
3. **Structured Data Set:** This module contains the structured data set for predicting student performance. The details are segmented into demographic factors, academic/work related factors social factors, personal factors and student academic result.
4. **Course Registration:** This module contains the comprehensive list of all the Course titles, Course codes, Credit Hour for the courses for the semester, the student is required to register his/her courses for the semester. Course registration is done semester by semester. Based on the curriculum and students area of specialty, the students selects courses his/her courses. The admin updates the curriculum when the need arises.
5. **Model Training/cross validation:** The prediction model is built from the training data using data mining technique. The system evaluates the model by using the prediction model with the processed training data, and used the training dataset for predicted student performance with actual student performance and evaluates the model by the Confusion Matrix and performance accuracy.
6. **Attributing Factors Module:** This module is designed in form a questionnaire and is used to factor in student demographic factors, academic and work related factors, social factors. Responses to these questions make up this module. Student's responses to the module is used for preprocessing as well as feature extraction for further analysis and to predict student academic performance
7. **Performance Predictor/ E-Advisor:** This module contains the prediction of student's performance. The system gets the predicted performance from the stored students' data set. The system displays the student's predicted performance with the suitable messages according to the predicted student's performance grade.
8. **Multi Agent Platform:**

- a. **User Interface Agent:** This agent is responsible for creating student as well as admin/teacher account. It is also the interface between the user and other agents It is also responsible for student personal data and their information such as feature/attributes variables, registration number, session, semester and other types of the data e.g. (demographic factors, academic related factors, social factors and other academic records).
- b. **Model Full Training Agent:** It is used for building the prediction model; this model is required to use the sample training data from user interface agent. The data will be trained with class label and stored in a required format for the classifier to use for performing the optimization results. The process will be done with the number of iteration and the result is displayed (Prediction Model) through the user agent.
- c. **Model Evaluation Agent:** This agent is used to evaluate student performance prediction; it receives the evaluation request from stored data with the required number of sample set: it contains the prediction model, test data set and the actual students' performance to evaluate the model and display the evaluation results to user interface.
- d. **Performance Prediction/Adviser Agent:** This agent requires student to input ID and then predict their performance. It also receives the prediction request from the User Interface Agent with the required test set data, prediction model and students' data to be predicted. These are stored in a required file format ready for prediction request from user agent. Student's registration number (ID) are used to predict their performance

#### 4.4.5 Input/output Format

Every program has an input as well as output data. These are used mainly to achieve the specific objectives of verifying the processing operation being performed. The lists of input and output forms that are available for the users to use in this new system.

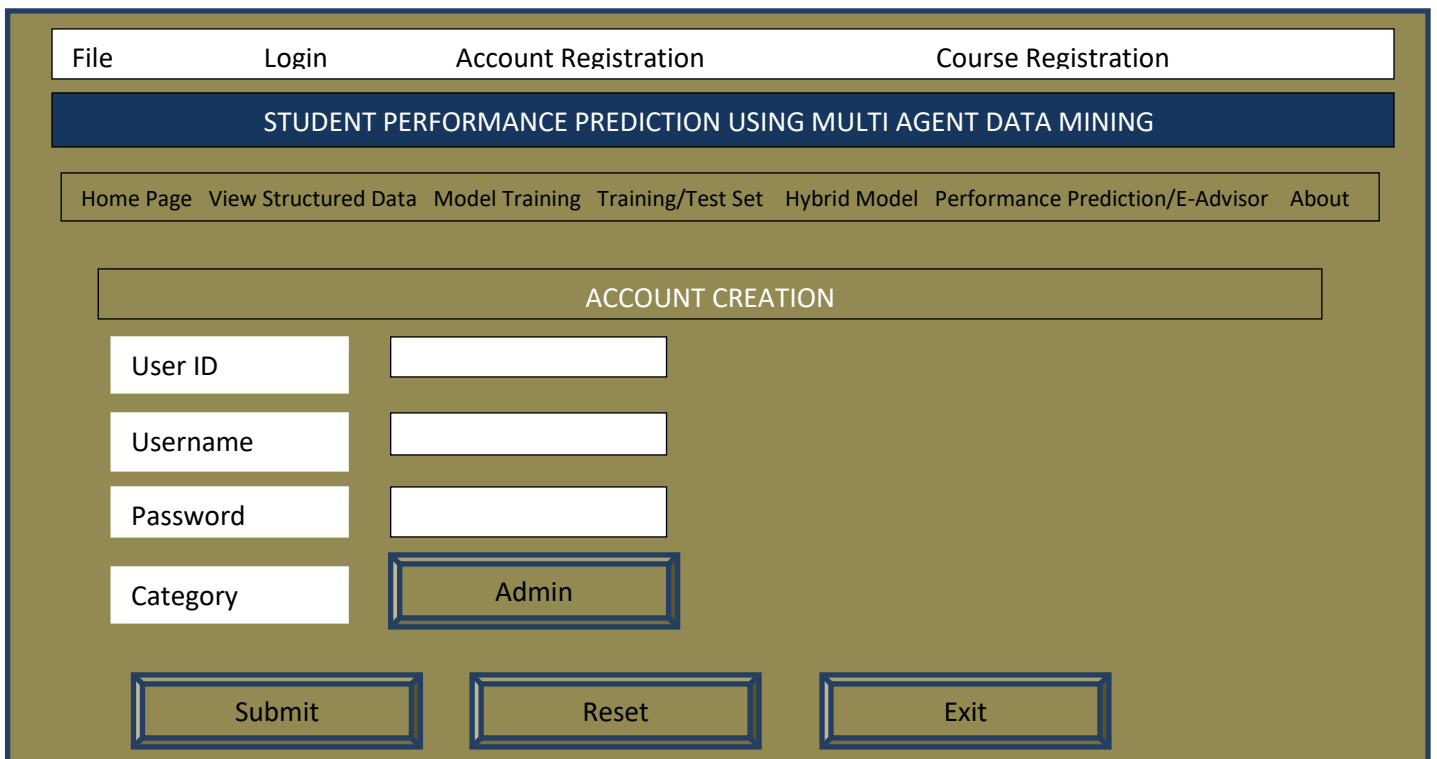
Figure 4.12 shows contain the login module for Admin, Lecturer and Student which includes the username and the password. Clicking on the login button will validate the data before launching the user on system and each user will be directed to their module.



The image shows a login module interface with a white background and a black border. It contains five blue buttons with white text: 'Login Panel' in the top left, 'Username' and 'Password' in the top right, and 'Login' and 'Forgot Password?' in the bottom center.

Figure 4.12: Login Module

Figure 4.13 shows Account Creation for Admin, Student or Lecturer. The admin is solely responsible for creating this account and specifying the category/ access level of the user. The user can thereafter change password on logging in.



The image shows a user account creation interface with a green background and a blue border. At the top, there is a navigation bar with 'File', 'Login', 'Account Registration', and 'Course Registration'. Below this is a dark blue header with the text 'STUDENT PERFORMANCE PREDICTION USING MULTI AGENT DATA MINING'. A secondary navigation bar contains 'Home Page', 'View Structured Data', 'Model Training', 'Training/Test Set', 'Hybrid Model', 'Performance Prediction/E-Advisor', and 'About'. The main content area is titled 'ACCOUNT CREATION' and contains four input fields: 'User ID', 'Username', 'Password', and 'Category'. The 'Category' field has a dropdown menu with 'Admin' selected. At the bottom, there are three buttons: 'Submit', 'Reset', and 'Exit'.

Figure 4.13: User Account Creation



Fig 4.14 shows the User Registration platform. This module is used to create account for newly admitted student and it is done by admin. Admin has to create account for each student before he/she can gain access to their semester course form. Information required in this the module include student ID, student registration number, session, and semester

User ID	Reg No	Year	Semester
0001	2016406085f	2017/2018	First Semester
0245	2017406005P	2017/2018	Second Semester

Figure 4.14: Student Registration Form

Figure 4.15 displays the Student Course Registration Module. The module displays the courses available for the selected semester in addition to their credit hour. The student selects fills his name and registration number and thereafter selects the session and semester. The courses for the selected semester are displayed. The student goes ahead to select his courses based on his area of specialty

**REGISTER YOUR FIRST SEMESTER COURSES**

Name

Reg No

Session

Semester

S/N	Select Course	Course Title	Credit
1	<input style="width: 100%; height: 20px;" type="text" value="ACC 811"/>	<input style="width: 90%; height: 20px;" type="text" value="Financial Accounting Theorv"/>	<input style="width: 20px; height: 20px;" type="text" value="3"/>
2	<input style="width: 100%; height: 20px;" type="text" value="ACC 831"/>	<input style="width: 90%; height: 20px;" type="text" value="Management Accounting Theorv"/>	<input style="width: 20px; height: 20px;" type="text" value="3"/>
3	<input style="width: 100%; height: 20px;" type="text" value="Select Course"/>	<input style="width: 90%; height: 20px;" type="text" value="Choose one Elective"/>	<input style="width: 20px; height: 20px;" type="text" value="3"/>
	<input style="width: 50px; height: 20px;" type="text" value="Save"/>	<input style="width: 100px; height: 20px;" type="text" value="Total Credit"/>	<input style="width: 50px; height: 20px;" type="text"/>

Figure 4.15 Student Course Registration Form

Figure 4.16 Displays the Home page. This module provides link to other module. It contains all the various task for user to use in accessing the system. It contains both main menu and submenus.

File
Login
Student/Admin Account Registration
Course Registration

**STUDENT PERFORMANCE PREDICTION USING MULTI AGENT DATA MINING**

Home Page
View Structured Data
Model Training
Training/test Set
Hybrid Model
Performance Predictor/E-Advisor
About

Figure 4.16: Home Page

Figure 4.17 displays the Structured Data Set Module. This module contains structured data set for predicting student performance. It is segmented into demographic factors, academic/work related factors social factors, personal factors and student academic result which forms the basis of the analysis.

STUDENT PERFORMANCE PREDICTION USING MULTI AGENT DATA MINING													
Home Page	View Structured Data	Model Training	Training/Test Set	Hybrid Model	Performance Predictor/E-Advisor	About							
Structured Dataset	DEMOGRAPHIC FACTORS			ACADEMIC/WORK RELATED FACTORS				SOCIAL FACTORS					
Reg No	Mode	Gender	Marital	Program	City	Age	Emp/Stat	Family Size	Sponsor	NAAC	Job Co	Attnd_Lect	
2014406008P	2	1	1	2	2	1	1	3	1	3	3	5	1
2017406085P	2	1	1	2	2	1	1	3	2	2	3	1	
2015406058F	1	2	2	2	1	2	2	2	2	5	5	5	5

Figure 4.17: Data Set

Figure 4.18 shows the Model Building interface, the model is built on both training set and testing test with machine learning algorithm, naïve Bayes and K-nearest neighbor. The training set is used for cross validation The entire data is divided into 10 subsets and each classifier is trained ten times excluding a single subset. The resulting classifier is then tested on the excluded subset. The result of the cross validation, the graphical representation of the analysis (pie chart and bar chart) of the algorithm is displayed, including the various performance evaluation metrics for each model.

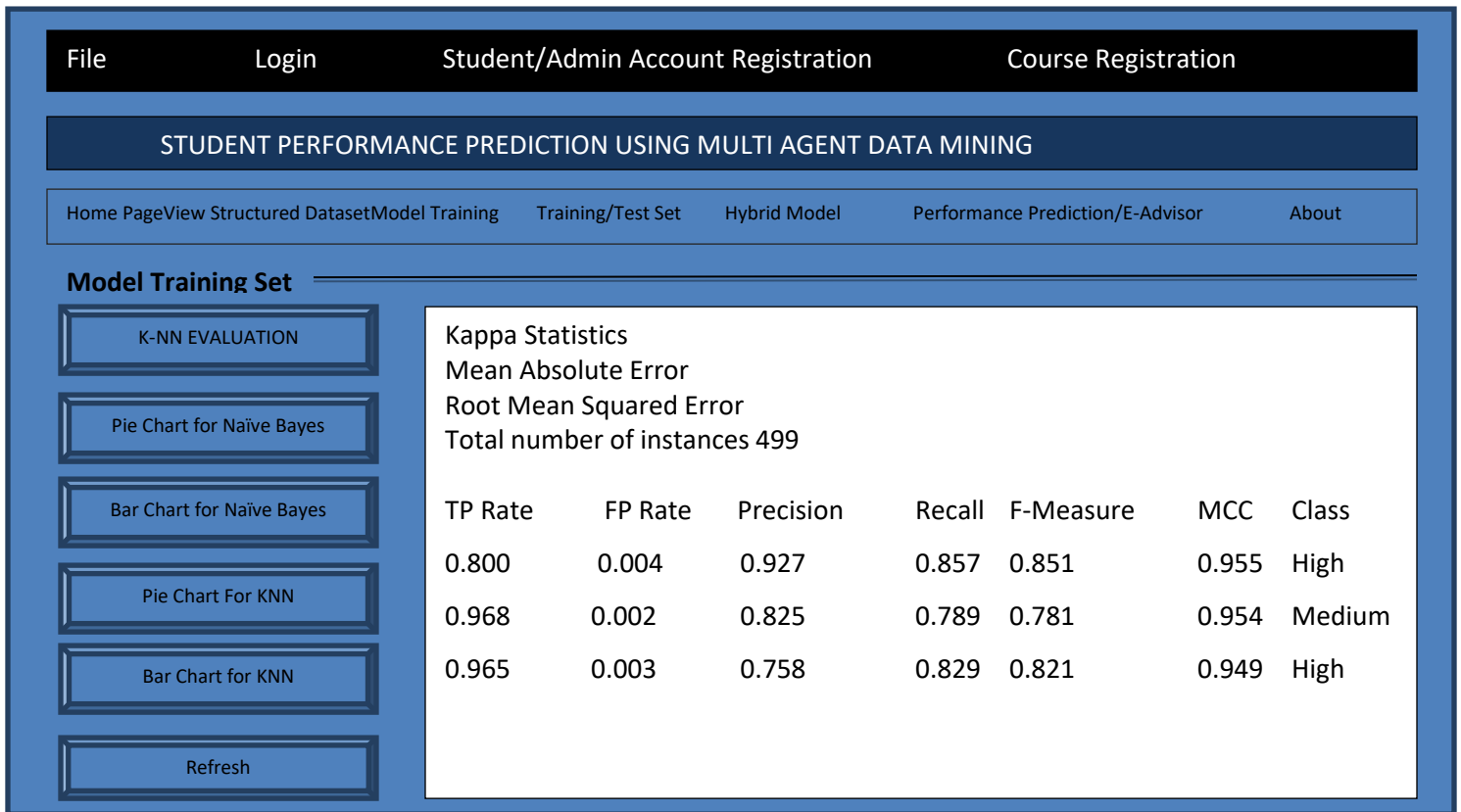


Figure 4.18 Model Building and Evaluation Interface

The entire system is again split into two sets; the training and test data using percentage split, in this case 90% training data and 10% test data was used. The system is trained using the two algorithms/classifiers. Evaluation of the model is done with the processed test data set. The dataset is used for the student performance prediction and evaluation is done using Confusion Matrix. The result of the analysis, the graphical representation of the analysis (pie chart and bar chart) of the algorithm is displayed; including the various performance evaluation metrics for each model is shown in figure 4.19.

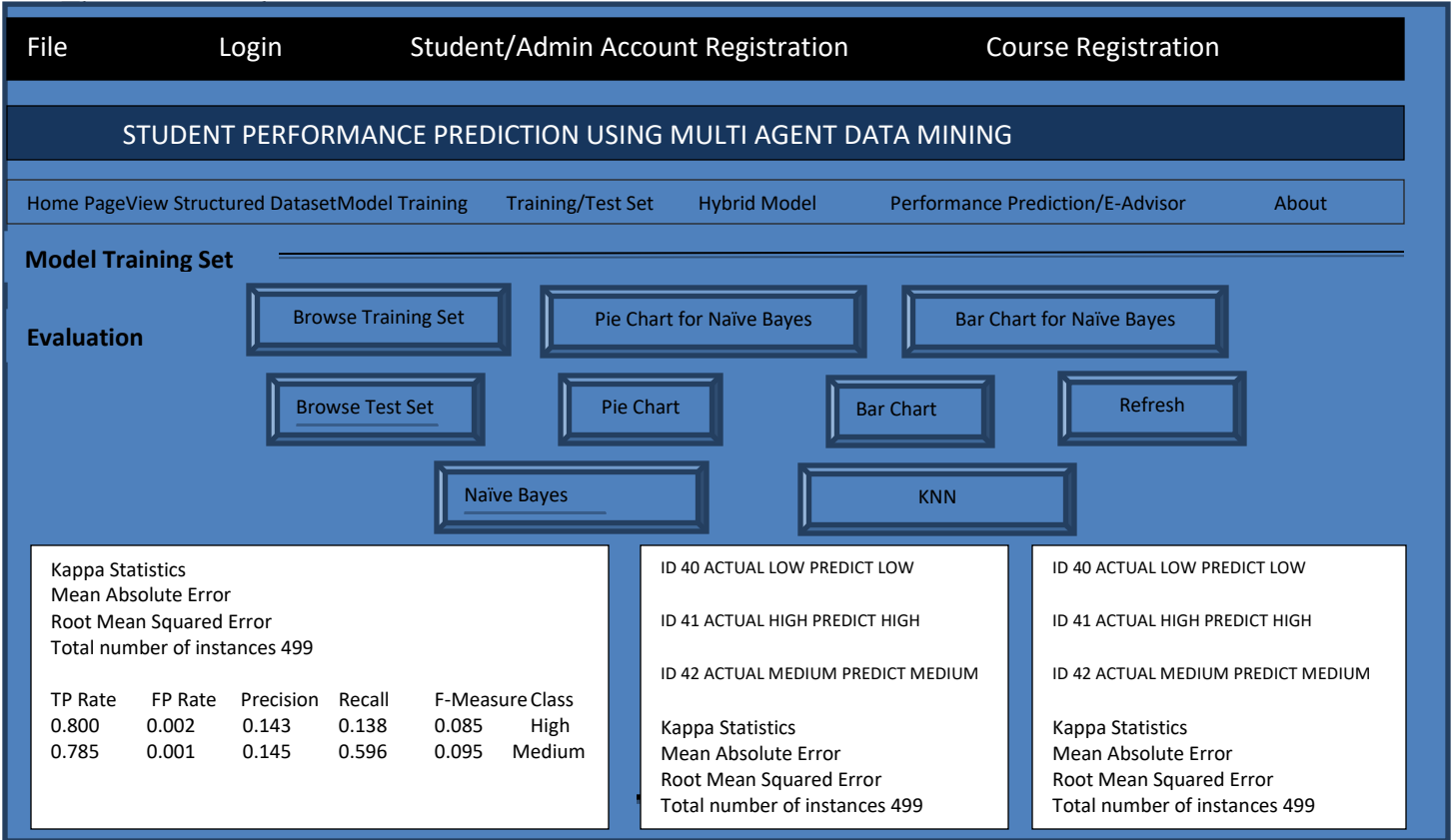


Figure 4.19 Training/Test Set

Figure 4.20 show the details performance prediction by student ID, the system gets the predicted performance from the stored students' data set. The system views the student's predicted performance with the suitable messages according to the predicted student's performance grade.

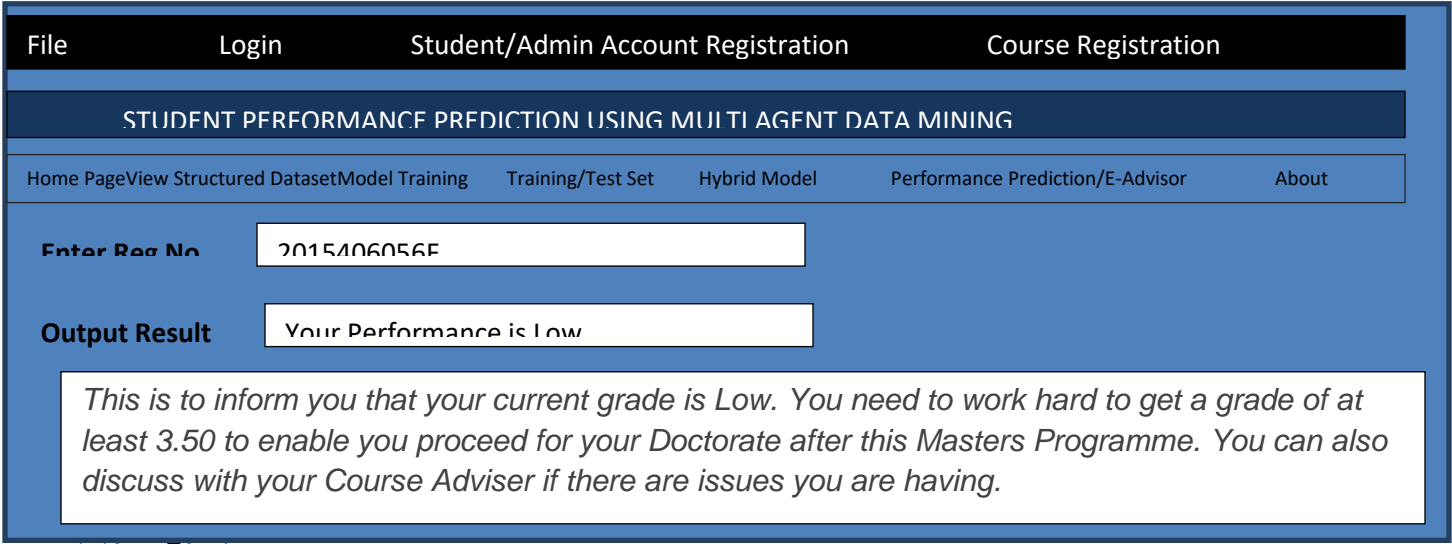


Figure 4.20 Students Performance Prediction

**Step1:** Let  $n$  be the number of instances in the training data  
 For each of  $t$  iterations do  
 Randomly sample  $n$  instances (using deletion and replication)  
 Apply a learning technique to build a model from the sample  
 Store the model  
 Make prediction from the model from test set  
**End**  
**Step2:** Assign equal weight to all instances  
 For each of  $t$  iterations do  
     Apply a learning technique to build a model from the weighted instances  
 and store the resulting model  
     Down-weight each instance correctly classified by the model  
**End**

**OR**

**Step 1- Start**

**Step 2-Take input which is given by User**

$In = \{I1, \dots, In\}$

**Step 3-Dataset preparation**

$Dn = \{ \{I1, \dots, In\} D \}$

**Step 4-Dataset elaboration**

$DI = \{S1, \dots, Sn, C1, \dots, Cn, I1, \dots, In, a1, \dots, an\}$

**Step 5- Processing**

While(  $Dn \neq 0$  )

{           If (  $an = In$  )

Check  $Cn, Sn$ ;

}

**Step 6- Result Generation**

$R = \{ Sc, Sn, Cn \}$ ;

Where,

$In$  = Input given by users

$Dn$  = Dataset

$D$  = Database

$DI$  = Dataset contents

$Sc$  = Semester Score

$a1, \dots, an$  = grade

$S1, \dots, Sn$  = Course

$C1, \dots, Cn$  = Category(High, Low and Medium)

#### 4.4.7 Data Dictionary

A data dictionary, or data repository, is a central storehouse of information about the system's data. The main purpose of a data dictionary is to describe, document and organize facts about the system and the database. The data dictionary for the system is illustrated in table 4.6

Table 4.6 Data Dictionary

<b>Variable Name</b>	<b>Meaning/Functions</b>
btnLogin	It is used to access both student/admin authentication during user login
btnExit	It is used to log out/exit from the system
btnSubmit	It used to submit student account during registration
btnReset	Its an option for resetting/refreshing student account
btnDemographicFactor	It is used to display student demographic factors
btnAcademic/workFactors	It is used to display student academic related factors
btnSocialFactors	It is used to display student social factors
btnNaiveBayesEvaluation	Display the result perfection of Naïve Bayes
btnK-NN	It is used to display the result perfection of K-NN
btnPieChatForNB	It is used to display pie chat for Naïve Bayes
btnBarChatForNB	It is used to display bar chat for Naïve Bayes
btnPieChatForK-NN	It is used to display pie chat for K-NN
btnBarChatForK-NN	It is used to display bar chat for K-NN
btnDataMiningClassifier	It is used to display the result of the selected data mining classifier
optClassifierOption	It is used for selecting the classification option
btnbrowse-Training set	It is used to browse the training set for classification model
btnbrowse-Test	It is used to browse the test set for classification model
btnROCFForNaiveBayes	It is used for the display the operational curve for Naïve Bayes
btnROCFForK-NN	It is used for the display the operational curve for K-NN
btnPie	It is a function used for the display pie chat of both naïve Bayes and K-NN
btnBar	Displays bar chat of both naïve Bayes and K-NN
btnPrediction	It is used for displaying prediction results of the model
btnSave	It is used for saving semester course form
btnDatasetStructure	It is used to view dataset structured

## 4.5 System Flowchart

System flowchart is the graphical representation of the flow of data in the system, and represents the work process of the system. Various symbols are used in the flowchart to designate specific actions. The new system flow diagram is shown in figure 4.21. Here the user login with his username and password to access the system. If login is successful the user can access other options available to him/her, if unsuccessful the user is redirected to the login page to enter the correct details. Depending on the access role the user navigates the options available and also request for performance prediction. With the user identification (registration number) the system predicts the students performance and offers academic advice based on the academic standing.

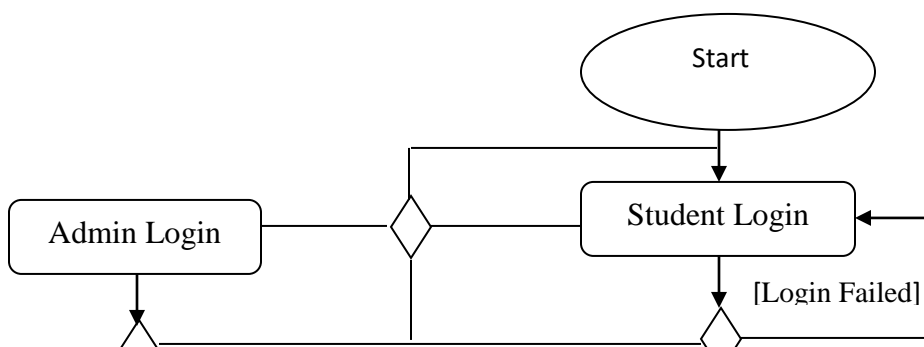
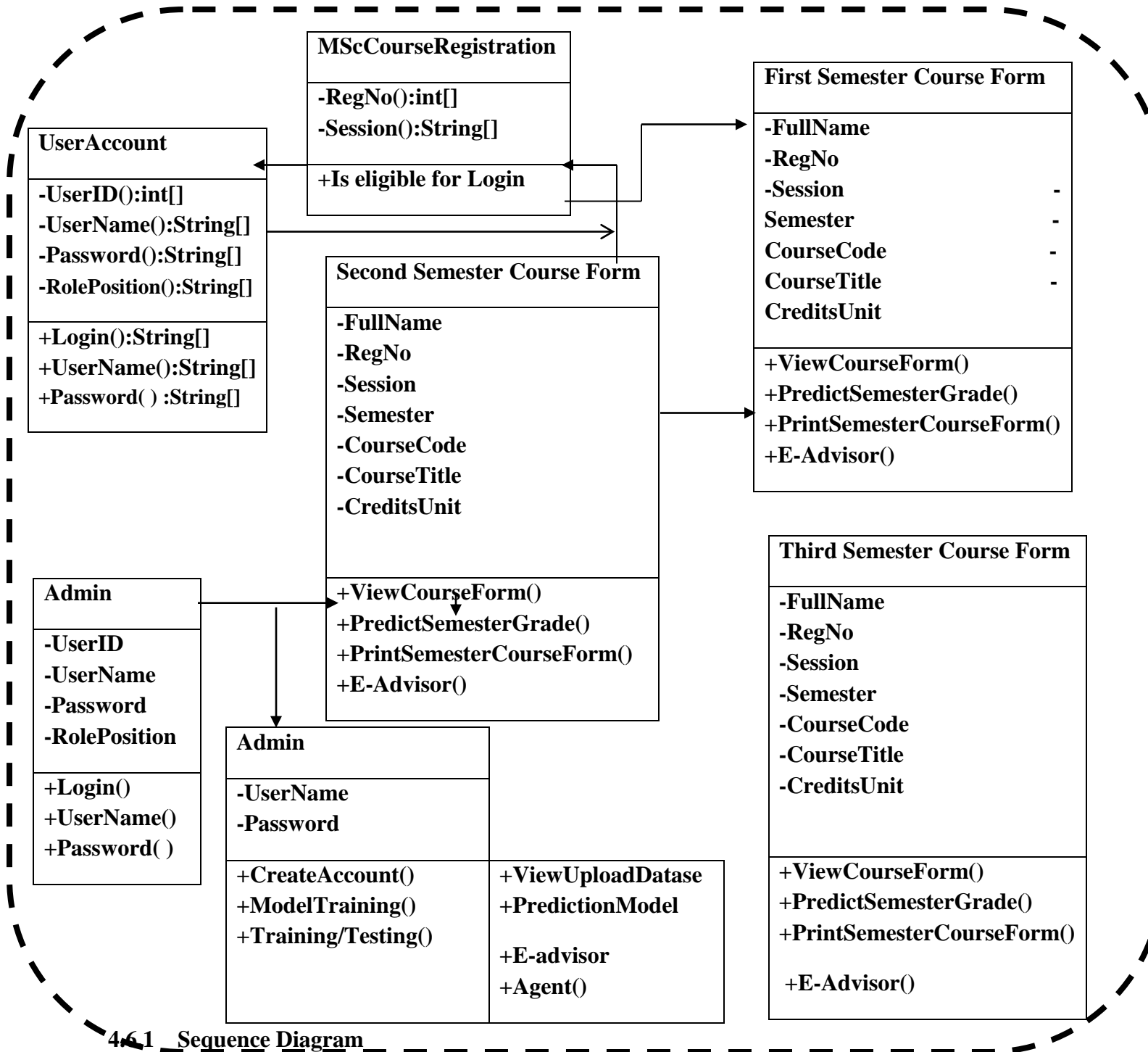




Figure 4.21 System Flow chart for Student's Performance Prediction Hybrid Model

## 4.6 Object Diagram

Object Diagram shows how objects in your system are interacting with each other at some point in time, and what values do they contain when the program is in a certain state. The object diagram for the new system is shown in figure 4.22.



4.6.1 Sequence Diagram

The flow of the major functions of the system is explained using the sequence diagram to show how objects interact in a given situation and how processes operate with another and in which order they operate. The major functions of this system need to answer the following questions as: Which type of users deal with it? Who manage the system users and assign roles? Who make the complaint? Who deal with each complaint and according to which criteria? Who solve the complaint? Who follows up each complaint?

An important characteristic of a sequence diagram is that time passes from top to bottom: the interaction starts near the top of the diagram and ends at the bottom. A popular use for them is to document the dynamics in an object-oriented system. For each key collaboration, diagrams that created show how objects interact in various representative scenarios for that collaboration.

Figure 4.23 describes the sequence diagram of the student performance prediction (a use case actor), the application system and the database. In Figure 4.23, the student logs into the system, the database checks for user authentication and grants the user access into the system. He can then register for semester course form, print he/her course form, and also predict his semester academic results. The performance determines the response from the e-advisor agent.

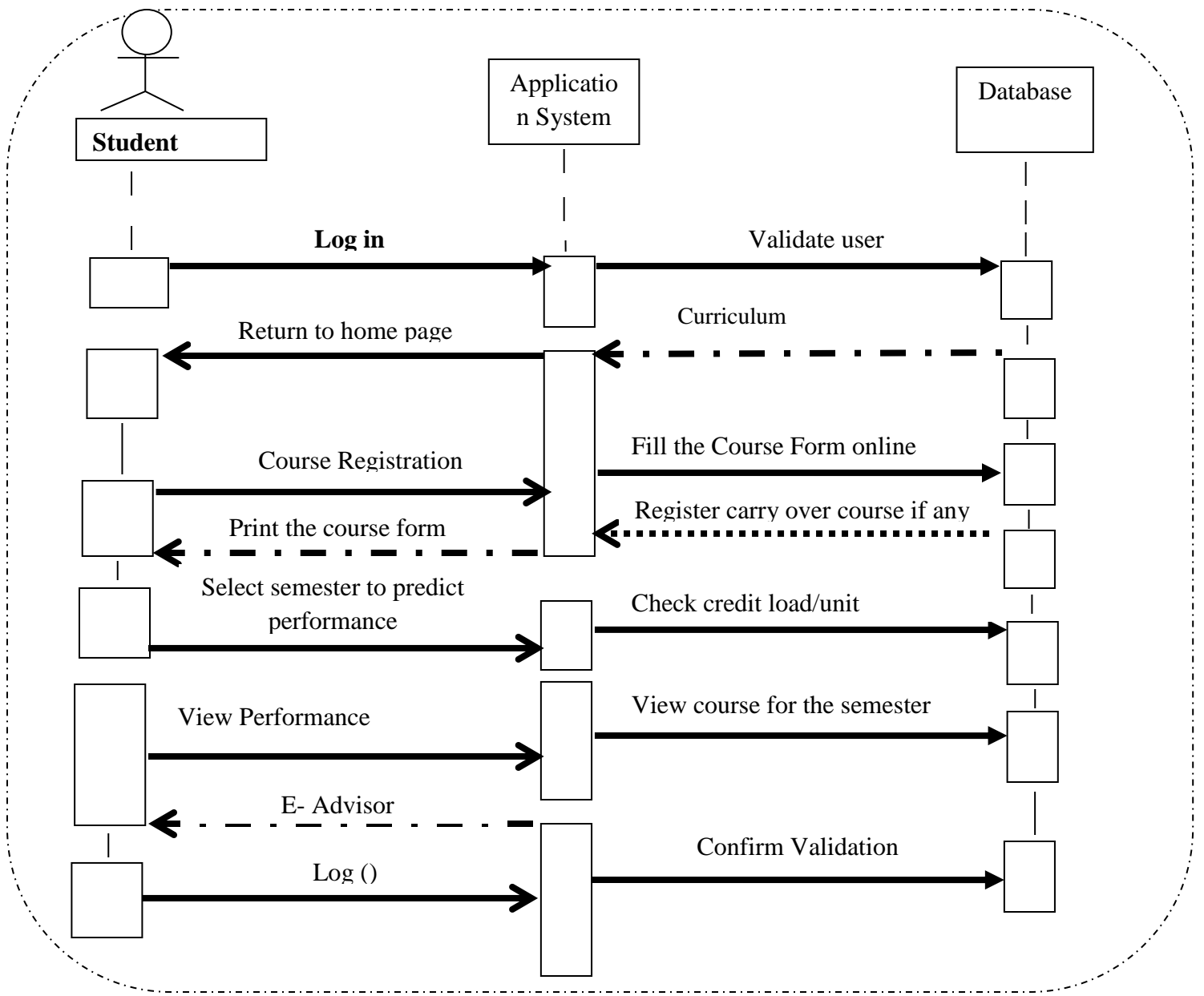


Figure 4.23: Sequence Diagram for Student

Figure 4.24 describes the sequence diagram of the student performance prediction (a use case actor), the application system and the database. The admin logs into the system, the database checks for user authentication and grants the user access into the system. He can update the curriculum, create account for users, request student's academic standing and also generate report on student's performance

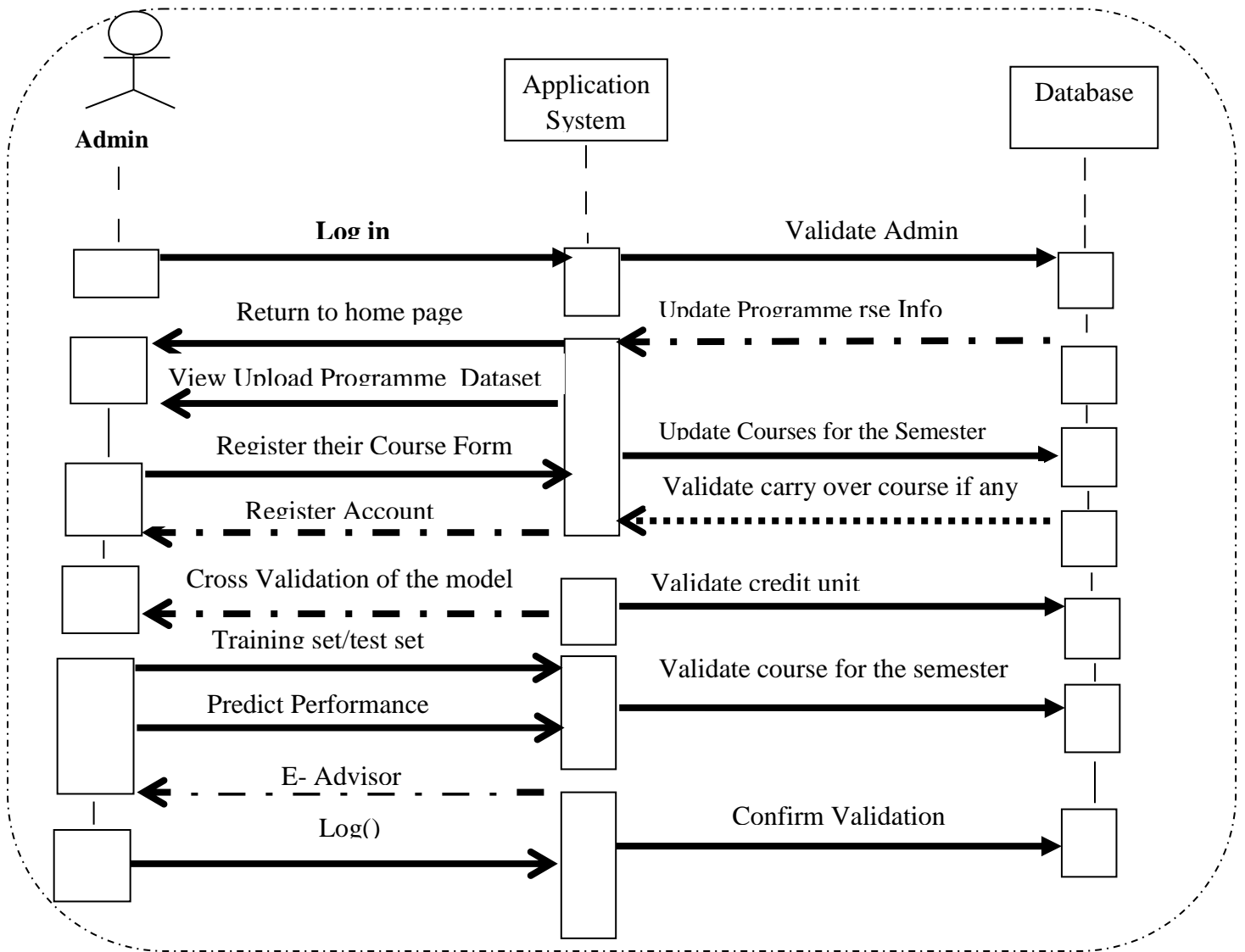


Figure 4.24 Sequence Diagram for Administrator

## 4.6.2 Activity Diagram

An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram. Activity diagrams are often used in business process modeling. They can also describe the steps in a use case diagram. Activities modeled can be sequential and concurrent.

The purpose of Activity diagram is to provide a view of flows and what is going on inside a use case or among several classes. We can also use activity diagrams to model code-specific information such as a class operation. Activity diagrams are very similar to a flowchart because you can model a workflow from activity to activity. It is basically a special case of a state machine in which most of the states are activities and most of the transitions are implicitly triggered by completion of the actions in the source activities.

Figure 4.25 shows the activity diagram of Postgraduate student navigating through the Student performance prediction with multi agent data mining techniques. The concept of activity diagram also provide the details of administrative activities as shown in figure 4.26 where admin has a major role to play before student can to have access to their account

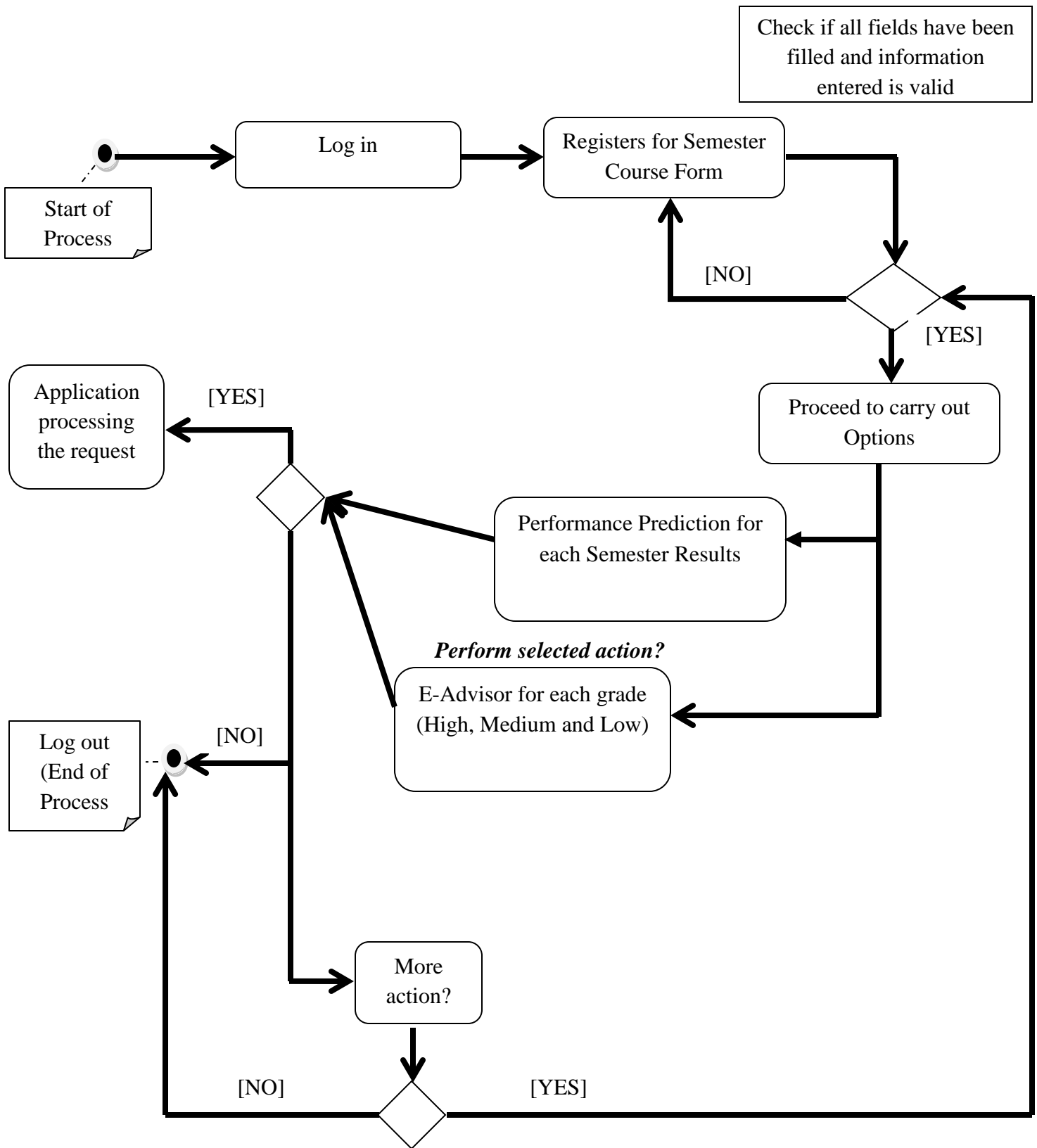


Fig 4.25: Activity diagram for Student

Check if all fields have been filled and information entered is valid

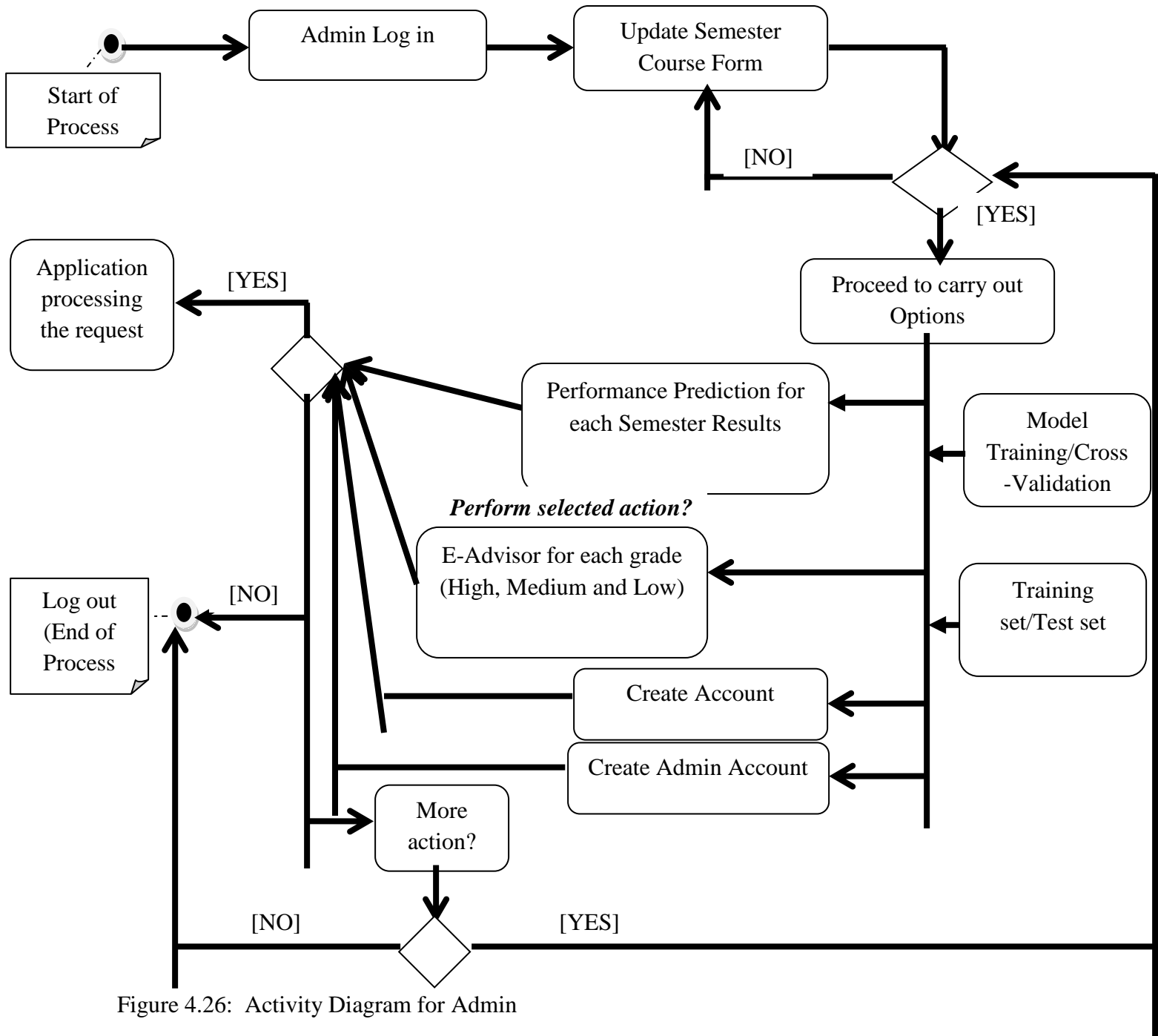


Figure 4.26: Activity Diagram for Admin



### **4.6.3 Class Diagram**

Class diagrams are the most popular UML diagrams used by the object oriented community. It describes the objects in a system and their relationships. Class diagram consists of attributes and functions. A single class diagram describes a specific aspect of the system and the collection of class diagrams represents the whole system. Basically the class diagram represents the static view of a system. The class diagram for the student performance prediction is shown in Figure 4.27

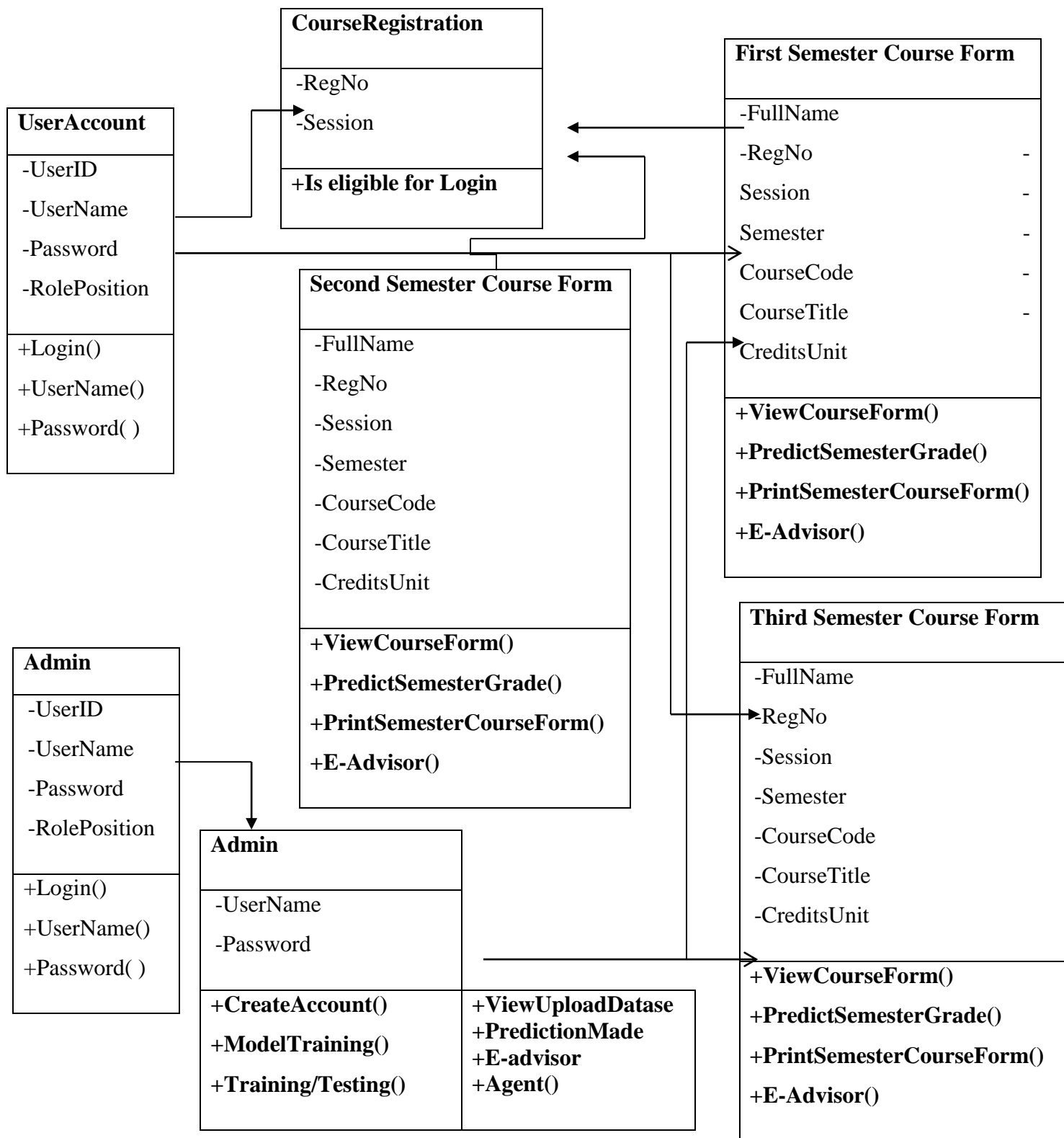


Figure 4.27 Class diagram for Student Performance Prediction

#### 4.6.4 Use Case Diagram

The use case model of the UML is used to specify the functionality of the system from the users' point of view and show the way the system and the users interact to achieve its stated functions and perform its goal. Figure 4.28 and Figure 4.29 shows the use case diagram for the Student and the administrator.

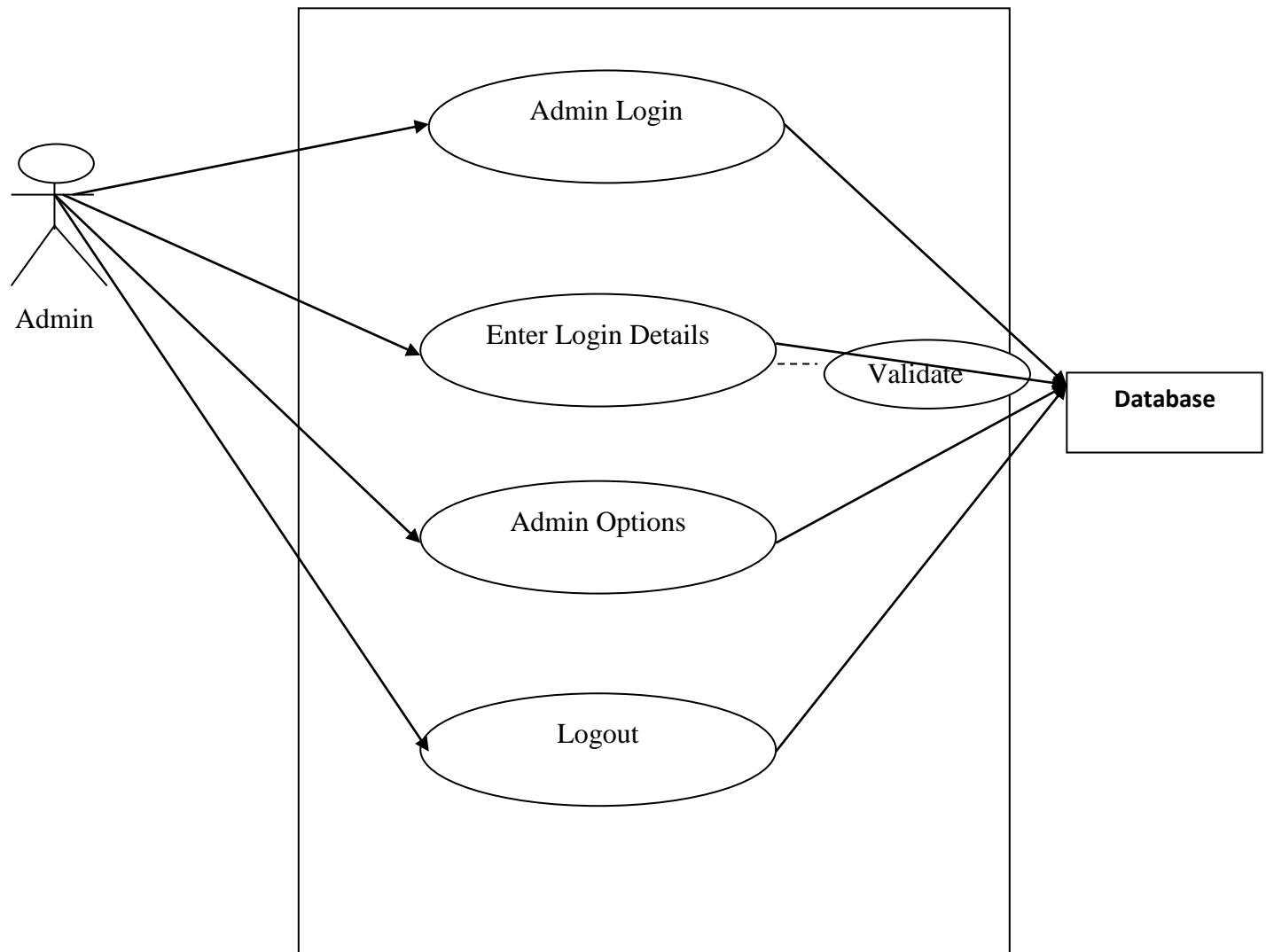


Figure 4.28: Use Case Diagram for Admin

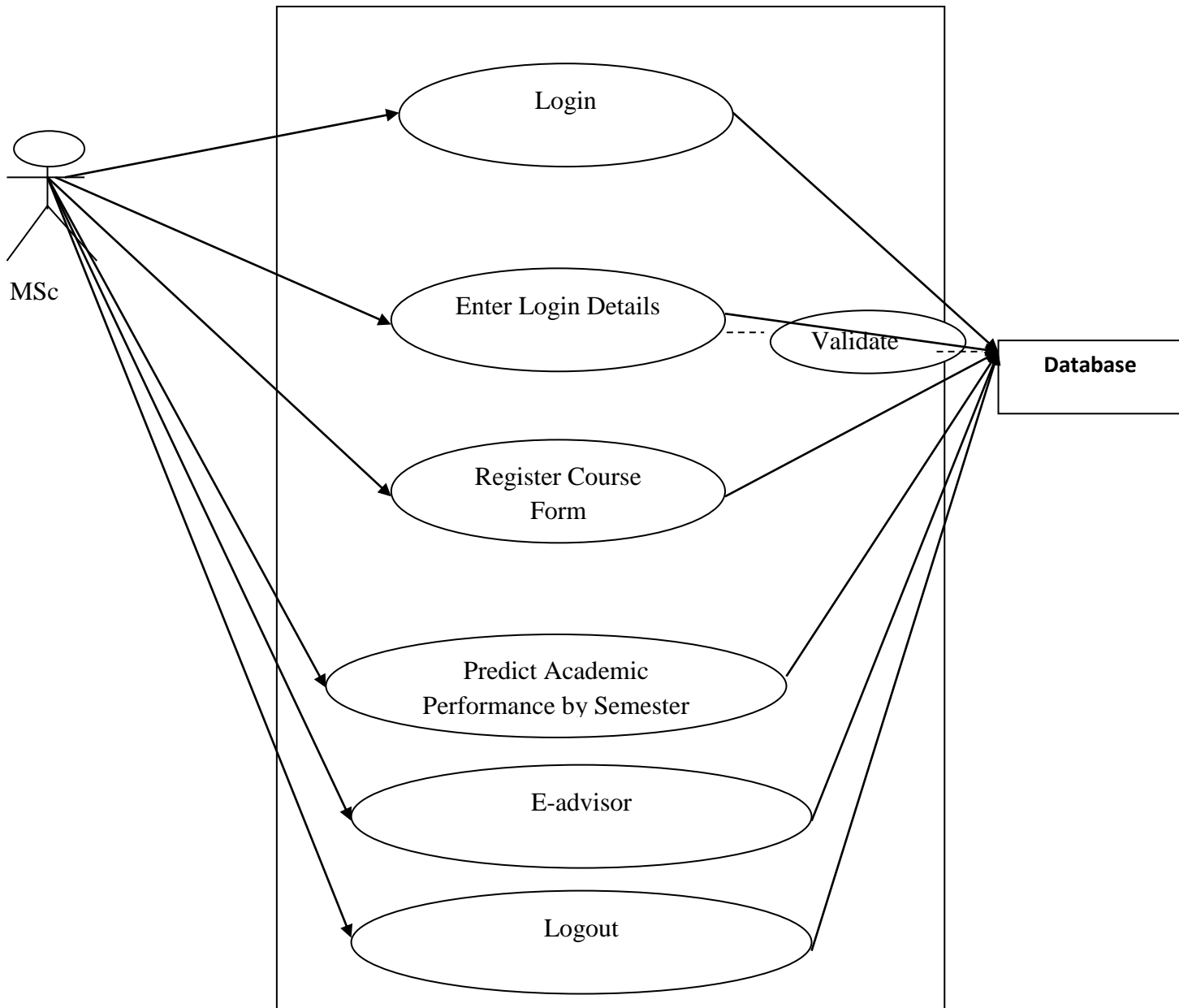


Figure 4.29: Use Case Diagram for Student



#### 4.6.5. Interaction Diagram

Interaction diagrams are models that describe how a group of objects collaborate in some behavior - typically a single use-case. The diagrams show a number of example objects and the messages that are passed between these objects within the use-case. The interaction diagram is shown in figure 4.31

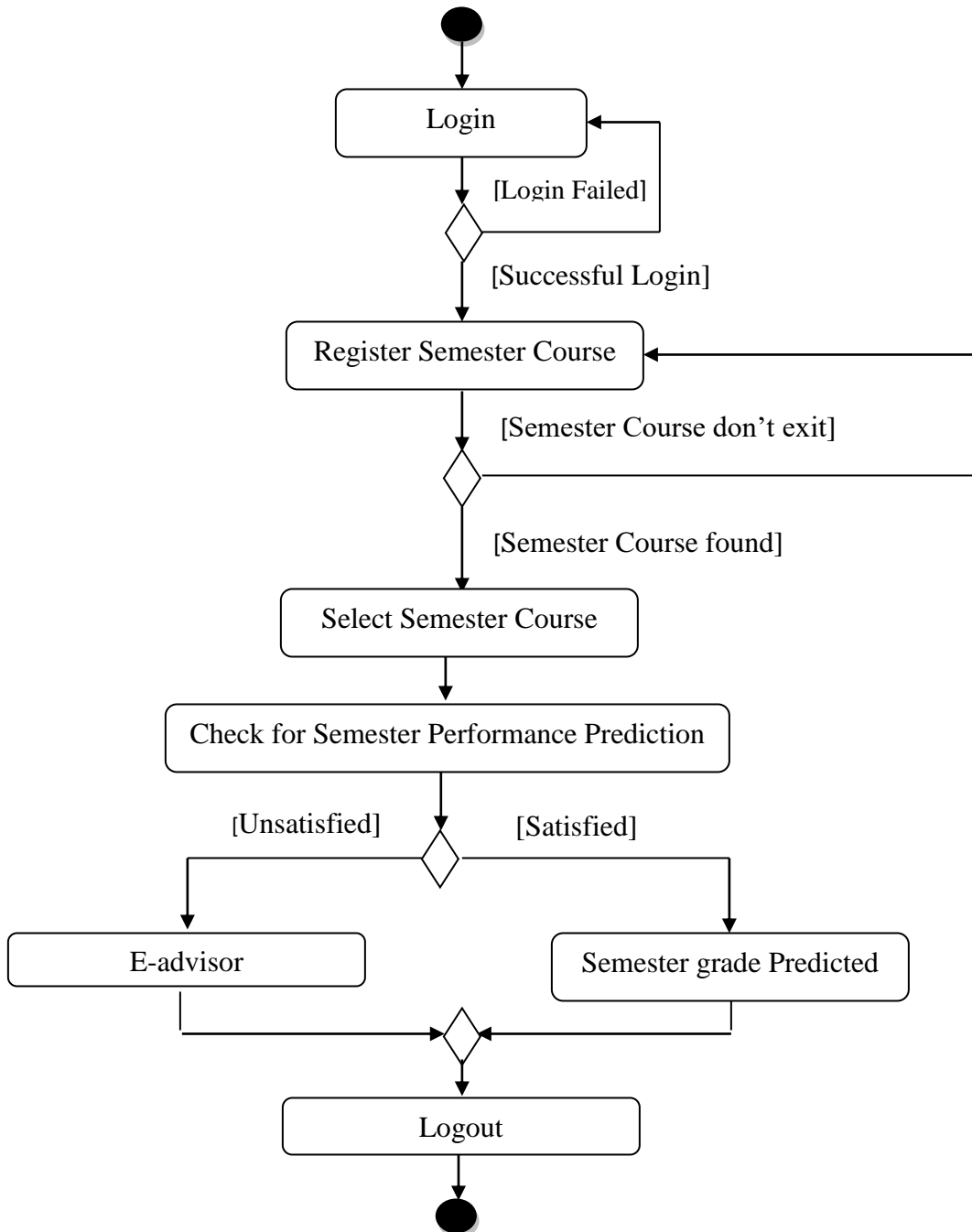


Figure 4.31: Interaction Diagram for Students' Performance Prediction

#### 4.6.6 Architecture of the new System

The architecture of the new system is shown in figure 4.32. The system is divided into three layers. The Admin/Teacher/ Student interface. Each user has his access level and what he/she can request or perform in the system.

The Data collection and preparation level involves collecting data from multiple sources and preprocessing to transform the data into the format that the algorithm can use to perform classification successfully. The third layer is the model building and evaluation using the single classifiers and the hybrid and the e-advisor module

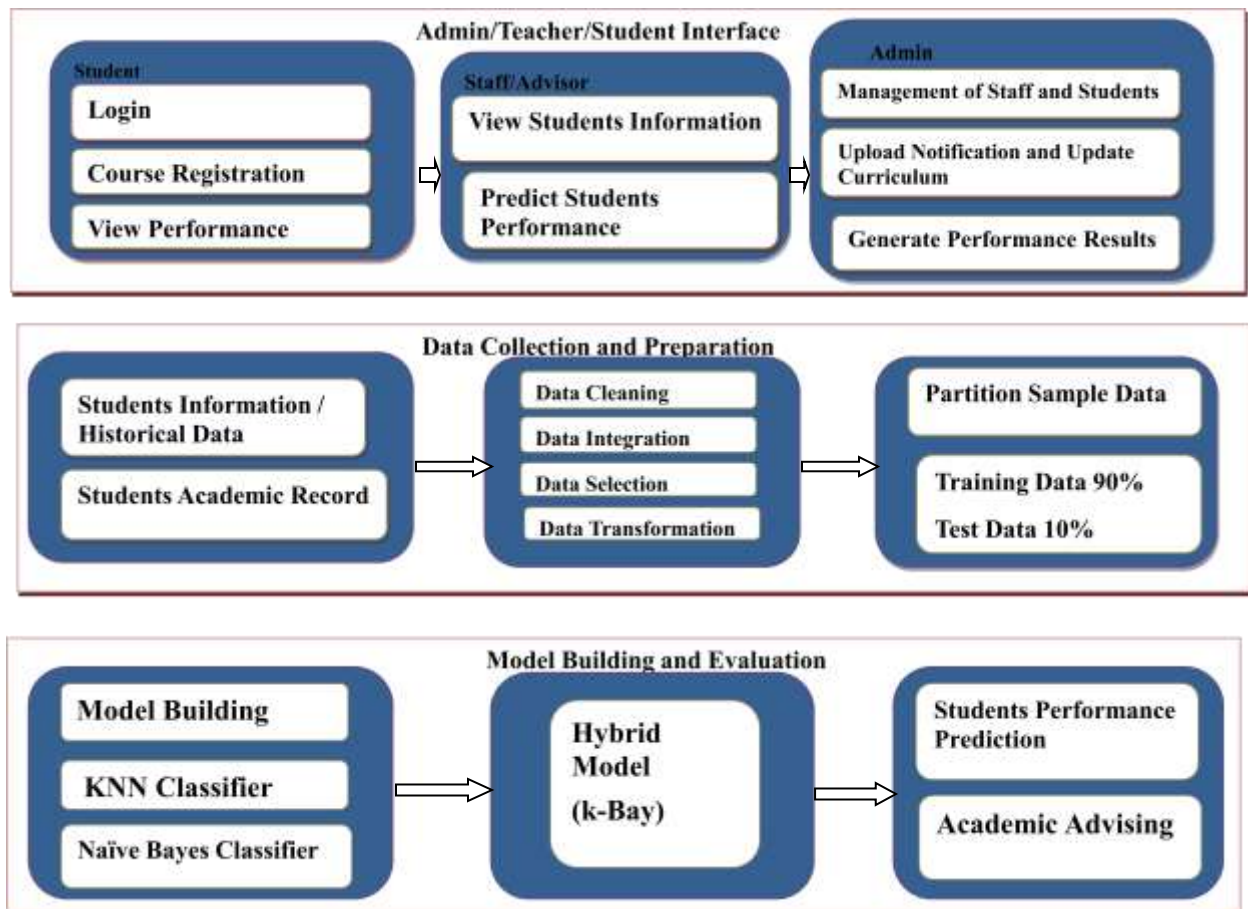


Figure 4.32: System Architecture of the new System

## **4.7 System Implementation**

In this research work, Java programming language (NetBeans IDE) was used to implement the system. The functionalities of the system were written using traditional java classes, these classes were then converted into an agent-based behavior. The implementation was done in Java programming language to take advantage of existing library JADE and WEKA. WEKA (Waikato Environment for Knowledge Analysis) is an open source toolkit, and it consists of a collection of machine learning algorithms for data mining tasks. Naïve Bayes and K-nearest neighbor algorithm was also used to evaluate the performance of the system. JADE is a platform that uses Java to establish multi-Agent system with a series of Agent behaviors. Agent communicates with other Agents by sending request messages and receiving the results.

### **4.7.1 Proposed System Requirements**

The computer system is divided into software and hardware. Both work together to achieve the desired goal in any application developed. In the developed system, the following are required:

#### **4.7.1.1 Hardware Requirements**

The following hardware components are needed

- ✓ Personal computer (PC) – a desktop or laptop
- ✓ Printer – a desk jet or inkjet
- ✓ Scanner – Preferably coloured scanner.

The personal computer should have a minimum of

- ✓ Core i3 Processor with processing speed of 1.2GHz or higher
- ✓ 4GB RAM
- ✓ 500 GB Hard disk

#### **4.7.1.2 Software Requirements**

- ✓ Installation of Java jdk into the system



- ✓ Java run time Environment (JRE )
- ✓ Netbeans IDE version 7.3.1 or latest
- ✓ Weka Run Time Environment (WRTE)
- ✓ Weka Tool version 3.8.1 ( Machine Learning Techniques)
- ✓ JADE (Java Agent Development Framework)
- ✓ Notepad++

## **4.7.2 Program Development**

The main processes involved in this research work are feature extraction, classification and prediction. For this purpose we use java programming.

### **4.7.2.1 Choice of Programming Environment**

The Students Performance Prediction System using data mining and multi-agents was developed using a combination of programming frameworks. For performing data pre-processing and feature extraction, Java Development Kit (JDK) and software development kit (SDK) were used. This contains a Java compiler, a full copy of the Java Runtime Environment (JRE), and many other important development tools. MySQL JDBC Driver and edu.mit.jwi\_2.1.4 are the two main libraries used for pre-processing data.

Net Beans IDE was used to develop for machine learning application using java programming language. It provides support for Java Development Kit 7. Net Beans Integrated Development Environment runs on the Java SE. Development Kit (JDK) which consists of the Java Runtime Environment plus developer tools was used for compiling, debugging, and running the applications written in the Java language as well as WEKA library. JADE(Java Agent Development Environment) library was used as a choice of programming environment.

### **4.7.2.2 Language Justification**

Java program is made up of many classes, objects and inheritance, each has its own program code, and each can be executed independently and at the same time each can be linked together in one way or another. Java is also simple and straightforward to use, secure and multithreaded

In Waikato Environment for Knowledge Analysis (Weka) a whole range of data preparation, feature selection and data mining algorithms are integrated. This means that only one data format is needed, and trying out and comparing different approaches becomes really easy. Weka comes with a graphical user interfaces GUI, which should make it easier to use. Weka is fully implemented in the Java programming language and thus runs on almost any modern computing platform. Weka also provides access to SQL databases using Java Database Connectivity and can process the result returned by a database query.

Netbeans is a powerful GUI builder; The IDE provides end-to-end solutions for all Java development platforms including the latest Java standards. NetBeans IDE has many integrated development modules, especially for Java. It has an easy to use Swing GUI design tool to build user interfaces through dragging and dropping components, such as Buttons and Textboxes. Its auto completed code gives programmers a list of possible code fragments to choose from, even as they type. It debugs and gives hints on optimizing code, and even inserts the right code

### **4.7.3 System Testing**

Before a system is put into operation, its components programs must be tested to make sure it works as a separate entity and when integrated. System testing removes bugs from individual programs and system application. The testing of this system was done with training data set and test data set. Test was also done in both Windows 7 and 8 Operating System and it worked effectively.

#### **4.7.3.1 Test Plan**

After completion of the detailed design, the design team will develop a plan for testing the software developed from the design specifications. Data sets must be generated as part of the test plan, which will effectively be used to test all functions of the software and its data outputs as defined in the specifications. Matrices of the test data that define the known values of all input data elements and the expected output values for each data element will be developed and documented. The matrices will also contain cross references to the design specifications for each

function to be tested. If new or unique data file and data structure is required for the testing, then two test data sets must be developed; one for use by the software developers for testing of their software and the other for the formal testing of the software under controlled conditions. The plan must cover all phases of the test to be performed including modular, integration, system, acceptance, and regression/classification testing.

**Unit/Module Testing:** This is a level of software testing where individual units/ components of software are tested. The purpose is to validate the performance of each unit of the software. The various units/modules have been tested and each has proved efficient as an entity.

**Integration testing:** This is done during and after integration of a new module into the main software package. This involves testing each individual code module. The essence of this intergration is to check how the modules work when they are intergrated into subsystem and in the main system. Inother words, the test carried out here is to ascertain that those modules do not loose their efficiency and reliability (which has been proved in the module testing above) when integrated into subsystem and system.

**Performance Testing:** Testing to assure that response times, run times, and other phases of execution are within acceptable limits and time frames

#### **4.7.3.2 Test Data**

Tests are conducted at all levels of the system development process by assigned review teams using approved test data plans and validated test data sets. Unit, modular, integration, system, acceptance, and classification data mining tests are conducted on all modules of the system. Testing is done during system development, system implementation and each time a change is subsequently made in the system

#### **4.7.3.3 Actual Test Result versus Expected Test Result**

The Student Performance Prediction using data mining and multi- agent was able to predict student's result and performance based on the rules set to predict the analysis. The information is then made accessible to the user via an interface that the user can use to further analyze the data.

Table 4.7 Actual Test Result versus Expected Test Result

<b>Module</b>	<b>Expected Test Result</b>	<b>Actual Test Result</b>
Home Page	Expected to see the page containing links to other modules	The home page displayed platform and contains all the links to the various modules in the credit card fraud detection system
Log In	Expected to see the Log In form so that users can log in.	When clicked on, a form appeared where the necessary details can be entered: username and password for admin, Teachers and student.
User Account Registration	When clicked on , it is expected to display the form for entering new account details	When the button was clicked on, the system displayed User ID, Reg No, Session, Semester and others
Student Registration Form	It is expected to allow the students to register his/her form	On successful login the course registration form with the available courses and credit hour is displayed. If login is unsuccessful the user isn't granted access to the page
Structured Data	It is expected to display the structured data set	The page displays the structured data set used for analyzing students' performance segmented into demographic factors, academic/work related factors, social factors, personal factors and students transcript

<p>Model Building and Evaluation</p>	<p>In this module, it is expected that the prediction model evaluates the performance prediction of students.</p>	<p>In this model, the performance prediction model is built from Training Data using data mining technique (KNN and Naïve Bayes. The Prediction model is also evaluated using the confusion matrix.</p>
<p>ID: 2015406035F ID: 2017406085F ID: 2015406085P ID: 2017406023P ID: 2015406098P ID: 2014406125F ID: 2014406026F ID: 2014406106F ID: 2017406064P ID: 2017406063P ID: 2015406117P ID: 2017406059P ID: 2015406123P ID: 2016406085F ID: 2014406113F</p>	<p>MEDIUM MEDIUM LOW LOW LOW MEDIUM MEDIUM LOW LOW MEDIUM MEDIUM LOW MEDIUM MEDIUM LOW LOW HIGH MEDIUM</p>	<p>MEDIUM  MEDIUM LOW LOW LOW MEDIUM MEDIUM LOW  LOW MEDIUM MEDIUM LOW LOW HIGH MEDIUM</p>

**4.7.3.4 Performance Evaluation**

Evaluation of classification algorithms is one of the key points in any process of data mining. The most common tools used in analyzing the results of classification algorithms applied are: confusion matrix, learning curves and receiver operating curves (ROC).

#### 4.7.3.4.1 Confusion Matrix

The confusion matrix or contingency table is a visualization tool commonly used to present performances of classifiers in classification tasks. It is used to show the relationships between real class attributes and that of predicted classes. The level of effectiveness of the classification model is calculated with the number of correct and incorrect classifications in each possible value of the variables being classified in the confusion matrix.

Table 4.8 describes the Contingency table. The true positives (TP) mean the correct classifications of the positive class A; true negatives (TN) are the correct classifications of the negative class B; false positives (FP) represent the incorrect classification of the negative class A into the positive class A, and false negatives (FN) are the incorrect classification of the positive Class B into the negative class B. Below illustrate mathematics equation for student performance prediction:

Table 4.8: Contingency Table

Contingency	Predicted Class		
Actual class	Class	Class A	Class B
	Class A	<b>True Positive (TP)</b>	<i>False Negative (FN)</i>
	Class B	<i>False Positive (FP)</i>	<b>True Negative (TN)</b>

- a. Predictive **accuracy** of the classifier measures the proportion of correctly classified

instances. 
$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN}$$

- b. **True Positive Rate (TPR or Recall or Sensitivity)**: is the fraction of positive cases predicted as positive. It measures the percent of actual positive class A that are correctly

classified. 
$$\text{Recall or Sensitivity} = \frac{TP}{TP+FN}$$

- c. **True Negative Rate (TNR or Specificity)**: is the fraction of negative cases that were correctly classified as negative. It measures the percent of actual negative class B that are correctly classified. **Specificity** =  $\frac{TN}{TN+FP}$
- d. **Positive Predictive Value (PPV)**: often called Precision, it is the percentage of the class predicted to be positive that were correct. **Precision** =  $\frac{TP}{TP+FP}$
- e. **False Negative Rate (FNR)**: is the fraction of positive cases that were incorrectly classified as negative. It is the percentage of positive example that were incorrectly classified =  $\frac{FN}{TP+FN} = 1 - TPR$
- f. **False Positive Rate (FPR)**: is the fraction of negative cases predicted as positive. The percentage of negative example that were incorrectly classified =  $\frac{FP}{TN+FP} = 1 - TNR$
- g. **F-Measure**: A measure that combines precision and sensitivity is the harmonic mean of the two parameters. F-Measure =  $2 \times \frac{Recall * Precision}{Recall + Precision}$
- h. **Error**: Indicates the proportion of cases classified incorrectly. Error = 1 - Accuracy
- i. **Matthew's Correlation Coefficient (MCC)**: is a measurement to measure the quality of a binary classification. MCC formulation is given for binary classification utilizing *true positives (TP)*, *false positives (FP)*, *false negatives (FN)*, and *true negatives (TN)* values as given below:

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

- j. **Cohen's Kappa Statistic,  $\kappa$** , is a measure of agreement between categorical variables X and Y. Basically Kappa statistic is a measure to show the agreement of prediction with the true results. The Kappa statistic varies from 0 to 1, where,
- ✓ 0 = agreement equivalent to chance.
  - ✓ 0– 0.20 = slight agreement.
  - ✓ 0.21 – 0.40 = fair agreement.
  - ✓ 0.41 – 0.60 = moderate agreement.
  - ✓ 0.61 – 0.80 = substantial agreement.
  - ✓ 0.81 – 0.99 = near perfect agreement.
  - ✓ 1 = perfect agreement.

The formula to calculate Cohen's kappa for two raters is:

$$k = \frac{O_a - E_a}{1 - E_a},$$

Where:  $O_a$  = the observed accuracy.

$E_a$  = the expected accuracy

- k. **Mean Absolute Error (MAE)** measures the average magnitude of the errors in a set of predictions, without considering their direction. It's the average over the test sample of the absolute differences between prediction and actual observation where all individual differences have equal weight.

$$\text{Mean Absolute Error} = \frac{1}{N} \sum_{i=1}^N (\text{Predicted}_i - \text{Actual}_i)$$

- l. **Mean Squared Error (MSE)** is quite similar to Mean Absolute Error, the only difference being that MSE takes the average of the square of the difference between the original values and the predicted values.

$$\text{Mean Squared Error} = \frac{1}{N} \sum_{i=1}^N (\text{Predicted}_i - \text{Actual}_i)^2$$

- m. **Root Mean Squared Error (RMSE)**: is a frequently used measure of the differences between values (sample and population values) predicted by a model and the values actually observed. The RMSE for your training and your test sets should be very similar if you have built a good model. If the RMSE for the test set is much higher than that of the training set, it is likely that you've badly over fit the data. The formula for calculating Root Mean Square Error is:

$$\text{Root Mean Squared Error} = \sqrt{\frac{\sum_{i=1}^N (\text{Predicted}_i - \text{Actual}_i)^2}{N}}$$



Both MAE and RMSE express average model prediction error in units of the variable of interest. Both metrics can range from 0 to  $\infty$  and are indifferent to the direction of errors. They are negatively-oriented scores, which mean lower values are better.

n. **ROC Curve:** Is a commonly used graph that summarizes the performance of a classifier overall possible thresholds. It is generated by plotting the True Positive (Y-axis) against the False Positive (X-axis) as you vary the threshold for assigning observations to a given Class.

a. **Analysis on True Positive (TP) Rate**

**The True Positive (TP) rate** is the proportion of examples which were classified as class x, among all examples which truly have class x, i.e., how much part of the class was captured.. In the confusion matrix, this is the diagonal element divided by the sum over the relevant row. In a multi class contingency table, the TP are those values on the diagonals as shown in figure 4.9.

Table 4.9: Contingency table /Confusion Matrix for multi classification (Cross validation)

Naïve Bayes		A	B	C	<-- classified as
	A	<b>24 (TP)</b>	0	6	a = HIGH
	B	0	<b>250 (TP)</b>	18	b = LOW
	C	2	19	<b>180 (TP)</b>	c = MEDIUM

$$\text{True Positive Rate (TPR) or Recall or Sensitivity} = \frac{TP}{TP+FN}$$

$$\text{For HIGH (Naïve bayes)} = \frac{24}{24+(0+6)} = 0.800$$

$$\text{For LOW(Naïve bayes)} = \frac{250}{250+(0+18)} = 0.933$$

$$\text{For MEDIUM (Naïve bayes)} = \frac{180}{180+(19+12)} = 0.896$$

b. **Analysis on False Positive (FP) Rate**

**The False Positive (FP)** rate is the proportion of examples which were classified as class x, but belong to a different class, among all examples which are not of class x. In the matrix, this is the column sum of class x minus the diagonal element, divided by the rows sums of all other classes;

$$\text{FPR} = \frac{FP}{TN+FP}$$

$$\text{For HIGH(Naïve bayes)} = \frac{2}{268+201} = 0.004$$

$$\text{For LOW (Naïve bayes)} = \frac{19}{30+201} = 0.082$$

$$\text{For MEDIUM(Naïve bayes)} = \frac{24}{30+268} = 0.081$$

### c. Analysis on Precision / Positive Predictive Value (PPV)

**Precision** is the proportion of the examples which truly have class x among all those which were classified as class x. In the matrix, this is the diagonal element divided by the sum over the relevant column

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\text{For HIGH(Naïve bayes)} = \frac{24}{24+(0+2)} = 0.923$$

$$\text{For LOW(Naïve bayes)} = \frac{250}{250+0+19} = 0.929$$

$$\text{For MEDIUM(Naïve bayes)} = \frac{180}{180+6+18} = 0.882$$

### d. Analysis on F-Measure

**The F-Measure** is simply a combined measure for precision and recall as represented by the following equation:

$$\text{F-Measure} = 2 \times \frac{\text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}}$$

$$\text{For HIGH(Naïve bayes)} = 2 \times \frac{0.923 * 0.800}{0.923 + 0.800} = 0.857$$

$$\text{For LOW(Naïve bayes)} = 2 \times \frac{0.929*0.933}{0.929*0.933} = 0.931$$

$$\text{For MEDIUM(Naïve bayes)} = 2 \times \frac{0.882*0.896}{0.882*0.896} = 0.889$$

**e. Analysis on MCC**

$$\text{MCC} = \frac{TP*TN - FP*FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

In this scenario, three classes needs to be classified: A, B, and C. You apply the above formulation to calculate MCC for multi-class case after calculating *TP*, *TN*, *FP*, and *FN* values for each class as shown below.

$$TP = TP_A + TP_B + TP_C$$

$$TN = TN_A + TN_B + TN_C$$

$$FP = FP_A + FP_B + FP_C$$

$$FN = FN_A + FN_B + FN_C$$

$$\begin{aligned} \text{MCC for High (Naïve Bayes)} &= \frac{TP*TN - FP*FN}{\sqrt{(TP+FN)(TN+FP)(TP+FP)(TN+FN)}} \\ &= \frac{24 * 467 - 6 * 2}{\sqrt{(24 + 2)(467 + 6)(24 + 6)(467 + 2)}} \\ &= \frac{11208 - 12}{\sqrt{(26)(473)(30)(469)}} \\ &= \frac{11196}{\sqrt{173032860}} \\ &= \frac{11196}{13154.19553} \\ &= 0.851 \end{aligned}$$

$$\begin{aligned} \text{MCC for Low (Naive Bayes)} &= \frac{250*212 - 18*19}{\sqrt{(250+19)(212+18)(250+18)(212+19)}} \\ &= \frac{53000 - 342}{\sqrt{(269)(230)(268)(231)}} \end{aligned}$$

$$= \frac{52658}{\sqrt{3830247960}}$$

$$= \frac{52658}{61888.99708} = 0.851$$

$$\text{MCC for Medium (Naïve Bayes)} = \frac{180 \cdot 274 - 21 \cdot 24}{\sqrt{(180+24)(274+21)(180+21)(274+24)}}$$

$$= \frac{49320 - 504}{\sqrt{(204)(295)(201)(298)}}$$

$$= \frac{48816}{\sqrt{(3604661640)}}$$

$$= \frac{48816}{60038.83443}$$

$$= 0.813$$

#### f. Analysis on Kappa Statistics for Naïve Bayes

$$k = \frac{O_a - E_a}{1 - E_a}$$

Where:  $O_a$  = the observed accuracy.

$E_a$  = the expected accuracy

$$O_a = \text{Total Number} \frac{\text{Total Number of TP}}{\text{Total}} = \frac{24+250+180}{499} = 0.90982$$

$$E_a = \left[ \frac{30}{499} \times \frac{26}{499} \right] + \left[ \frac{268}{499} \times \frac{269}{499} \right] + \left[ \frac{201}{499} \times \frac{204}{499} \right] = (0.45733)$$

$$k = \frac{O_a - E_a}{1 - E_a}$$

$$k = \frac{0.90982 - 0.45733}{1 - 0.45733} = 0.8338$$

#### 4.7.3.4.2 Ranking of Attributes

Using correlation based feature selection technique on Weka, the attributes/ features were ranked to give the most important features with their descending order. Five attributes were ranked most important. They include: Employment-Status, Supervisor being too busy with extensive

commitment, Poor library facilities, standard equipment and Laboratory, Use of stimulants, drugs to enhance study, regular-use-of-the-internet-for-surfing-and-social-networking.

Result of the ranking is shown below:

=== Attribute Selection on all input data ===

Search Method:

Best first.

Start set: no attributes

Search direction: forward

Stale search after 5 node expansions

Total number of subsets evaluated: 311

Merit of best subset found: 0.412

Attribute Subset Evaluator (supervised, Class (numeric): 36 Encouragement-from-Sponsors-Partner-in-Postgraduate-Pursuit):

CFS Subset Evaluator

Including locally predictive attributes

Selected attributes: 6,15,30,34,35 : 5

Employment-Status

Supervisor-is-too-busy-with-extensive-commitment

Poor-library-facilities-Standard-equipment-and-Laboratory

Use-of-stimulants-drugs-enhance-study

Regular-use-of-the-internet-for-surfing-and-social-networking

#### **4.7.3.4.3 Results of the Performance Evaluation**

Classification and performance evaluation of the classifiers was done in two parts.

1. Performance evaluation using All the attributes.

- Performance evaluation using the highly influencing attributes as ranked using coefficient correlation based feature selection.

#### 4.7.3.4.3.1 Performance evaluation using All the attributes

Table 4.10 to table 4.16 shows the performance metrics using all the attributes and the exam scores.

Table 4.10: Contingency Table for Naïve Bayes and K-NN Evaluation (Cross Validation)

		A	B	C	<-- classified as	
Naïve Bayes	A	24	0	6	a =HIGH	Predicted Class by Naïve Bayes classifier
	B	0	250	18	b =LOW	
	C	2	19	180	c =MEDIUM	
K-NN	A	4	5	21	a =HIGH	Predicted Class by K-NN classifier
	B	6	180	82	b =LOW	
	C	18	66	117	c =MEDIUM	

Table 4.11 shows the result on the analysis from the 10 folds validation. The entire data set was used for the cross validation/ 90.98% were correctly classified by Naïve bayes while 60.32% were correctly classified using KNN algorithm.

Table 4.11: Filtered result for KNN and Naïve Bayes (Cross Validation using all attributes)

Classifier	Train (499 instances) Cross Validation using all attributes				
	Correctly Classified Instances Student	Incorrectly Classified Instances Student	Accuracy	Kappa statistic	Mean absolute error
Naïve Bayes	90.982%	9.018%	81.76%	0.8338	0.0834
K-NN	60.3206 %	39.6794 %	53.91%	0.2771	0.2657

Table 4.12 shows the metrics result for the two classifiers using cross validation with their various accuracy measures. Performance metrics like Precision, Recall, F-Measure, Mathews Correlation Coefficient (MCC), and Receiver Operator Characteristics (ROC) were used for the analysis.

Table 4.12: Naïve Bayes and K-NN Performance metrics results using all attributes

Algorithm	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
Naïve Bayes	0.800	0.004	0.923	0.800	0.857	0.851	0.993	0.937	HIGH
	0.933	0.082	0.929	0.933	0.931	0.851	0.968	0.973	LOW
	0.896	0.081	0.882	0.896	0.889	0.813	0.954	0.930	MEDIUM
K-NN	0.133	0.051	0.143	0.133	0.138	0.085	0.488	0.067	HIGH
	0.672	0.307	0.717	0.672	0.694	0.363	0.683	0.661	LOW
	0.582	0.346	0.532	0.582	0.556	0.234	0.614	0.477	MEDIUM

Using a 90% training data and 10% test data percentage split, the system was trained and tested using Naïve Bayes and KNN algorithm. The result of the analysis as shown in figure 4.13 shows the contingency table for the result for the percentage split for the training test and test set using the classifiers while table 4.14 shows the prediction accuracy and computational results.

4.13 Contingency Table of the result (Model Building and Evaluation using all attributes)

Classifier	A	B	C	<-- classified as		Predicted Class by Naïve Bayes classifier
	Naïve Bayes	22	0	2	a = HIGH	
	0	235	12	c = MEDIUM		
	5	19	154			
K-NN	14	1	9	a = HIGH	b = LOW	Predicted Class by K-NN classifier
	0	216	31	c = MEDIUM		
	8	33	137			

Table 4.14: Prediction Accuracy and Computational Results using all attributes

Classifier	Train set (449 instances)					Test set(49 instances)				
	Correctly Classified Instances Student	Incorrectly Classified Instances Student	Accuracy	Kappa statistic	Mean absolute error	Correctly Classified Instances Student	Incorrectly Classified Instances Student	Accuracy	Kappa statistic	Mean absolute error
Naïve Bayes	91.5367 %	8.4633 %	91.54	0.8425	0.0718	71.4286%	28.5714%	71.43%	0.4503	0.3965
K-NN	81.7372 %	18.2628 %	81.74	0.6585	0.1236	69.3878%	30.6122%	69.39%	0.4258	0.4381

Table 4.15 shows the metrics result for the two classifiers using percentage split with their various accuracy measures. Measures like Precision, Recall, F-Measure, Mathews Correlation Coefficient (MCC), and Receiver Operator Characteristics (ROC) were used for the analysis.

Table 4.15: Naïve Bayes and Performance metrics results using all attributes

Algorithm	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
Naïve Bayes	0.917	0.012	0.815	0.917	0.863	0.856	0.996	0.929	HIGH
	0.951	0.094	0.925	0.951	0.938	0.861	0.973	0.979	LOW
	0.865	0.052	0.917	0.865	0.890	0.822	0.963	0.946	MEDIUM
K-NN	0.583	0.019	0.636	0.583	0.609	0.588	0.773	0.415	HIGH
	0.874	0.168	0.864	0.874	0.869	0.707	0.847	0.813	LOW
	0.770	0.148	0.774	0.770	0.772	0.623	0.807	0.717	MEDIUM

Table 4.16 describes the performance evaluation of our proposed hybridization of Naïve Bayes and k-nearest neighbor. The result computed show the level of accuracy in each of the data mining classifier with Naïve Bayes having an accuracy of 71.43%, K-NN, is 69.39% and K-Bay is 95.92% respectively. Model Testing time for K-Bay was lower compared to KNN and Naïve Bayes. This result shows that the hybrid models gave better performance compared to the single model.

Table 4.16 K-Bay performance Evaluation using all attributes

Classifier	Test set (49 instances)					
	Correctly Classified Instances Student	Incorrectly Classified Instances Student	Accuracy	Kappa statistic	Mean absolute error	Model Testing Time (Sec.)
Naïve Bayes	71.4286%	28.5714%	71.43%	0.4503	0.3965	0.357
K-NN	69.3878%	30.6122%	69.39%	0.4258	0.4381	0.18
K-Bay	95.9184 %	4.0816 %	95.92%	0.9213	0.1395	0.134

#### 4.7.3.4.3.2 Performance Evaluation using highly Influencing Factors/attributes

Performance results using only the highly influencing attributes using only highlyinfluencing factors/attributes and the exam score is shown in Table 4.17 to table 4.23



Table 4.17: Naïve Bayes and K-NN Performance results using highly influencing attributes

Classifier	Train (499 instances) using Ranking (5 attributes)				
	Correctly Classified Instances Student	Incorrectly Classified Instances Student	Accuracy	Kappa statistic	Mean absolute error
Naïve Bayes	98.1964	1.8036 %	98.40%	0.9667	0.0558
K-NN	72.5451 %	27.4549 %	72.55%	0.4918	0.1848

Table 4.18: Naïve Bayes and K-NN Performance results using highly influencing attributes

Algorithm	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
Naïve Bayes	0.800	0.000	1.000	0.800	0.889	0.889	0.999	0.987	HIGH
	0.989	0.000	1.000	0.989	0.994	0.988	0.999	0.999	LOW
	1.000	0.030	0.957	1.000	0.978	0.963	0.997	0.995	MEDIUM
K-NN	0.233	0.032	0.318	0.233	0.269	0.233	0.608	0.145	HIGH
	0.813	0.216	0.813	0.813	0.813	0.597	0.795	0.760	LOW
	0.682	0.242	0.656	0.682	0.668	0.437	0.719	0.581	MEDIUM

4.19 Contingency Table of the result (Cross Validation using highly influencing attributes)

Classifier	A	B	C	<-- classified as		Predicted Class by Classifier
	Naïve Bayes	0	6	24	a =	
	265	3	0	b =	LOW	
	0	201	0	c =	MEDIUM	
K-NN	0	23	7	a =	HIGH	
	218	49	1	b =	LOW	
	50	137	14	c =	MEDIUM	

Table 4.20: Prediction Accuracy and Computational Results using highly influencing attributes

Classifier	Train set (449 instances)					Test set(49 instances)				
	Correctly Classified Instances Student	Incorrectly Classified Instances Student	Accuracy	Kappa statistic	Mean absolute error	Correctly Classified Instances Student	Incorrectly Classified Instances Student	Accuracy	Kappa statistic	Mean absolute error
Naïve Bayes	96.882 %	3.118 %	96.88	0.944	0.0615	75.5102 %	24.4898 %	75.51%	0.5859	0.1771
K-NN	76.8374 %	23.1626 %	76.84	0.5824	0.156	59.1837%	40.8163 %	59.18%	0.3436	0.2826

Table 4.21: Naïve Bayes and Performance metrics results using highly influencing attributes

Algorithm	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
Naïve Bayes	0.871	0.007	0.900	0.871	0.885	0.877	0.991	0.954	HIGH
	0.983	0.014	0.987	0.983	0.985	0.969	0.998	0.998	LOW
	0.968	0.031	0.958	0.968	0.963	0.936	0.991	0.981	MEDIUM
K-NN	0.677	0.029	0.636	0.677	0.656	0.630	0.886	0.543	HIGH
	0.857	0.247	0.785	0.857	0.819	0.614	0.825	0.782	LOW
	0.676	0.146	0.770	0.676	0.720	0.542	0.792	0.710	MEDIUM

4.22 Contingency Table (Model Building and Evaluation using highly influencing attributes)

Naïve Bayes	A	B	C	<-- classified as	Predicted Class by Naïve Bayes classifier
	<b>0</b>	4	27	<b>a =</b>	
226	<b>4</b>	0	<b>b =</b>	LOW	
3	182	<b>3</b>	<b>c =</b>	MEDIUM	
K-NN	<b>3</b>	7	21	<b>a =</b>	HIGH
197	<b>31</b>	2	<b>b =</b>	LOW	
51	127	<b>10</b>	<b>c =</b>	MEDIUM	
					Predicted Class by K-NN classifier

Table 4.23 describes the performance evaluation of our proposed hybridization of Naïve Bayes and k-nearest neighbor using only the highly influencing factors/attributes. The result computed show the level of accuracy in each of the data mining classifier with Naïve Bayes having an accuracy of 75.51%, K-NN, is 59.38% and K-Bay is 99% respectively. Model Testing time for K-Bay was lower compared to KNN and Naïve Bayes. From the result, KNN performed better when all the attributes were used while Naïve Bayes performed better when only the highly influencing attributes were used. The hybrid model also gave better performance when the only highly influencing attributes were used as compared to when all the attributes were used. .

Table 4.23 K-Bay performance Evaluation using all attributes using highly influencing attributes

Classifier	Test set (49 instances)				
	Correctly Classified Instances Student	Incorrectly Classified Instances Student	Accuracy	Kappa statistic	Mean absolute error
Naïve Bayes	75.5102 %	24.4898 %	75.51%	0.5859	0.1771
K-NN	59.1837%	40.8163 %	59.18%	0.3436	0.2826
K-Bay	99%	1%	99%	0.9999	0.0019

**4.7.3.4.3 Analysis on the various Factors/Attributes**



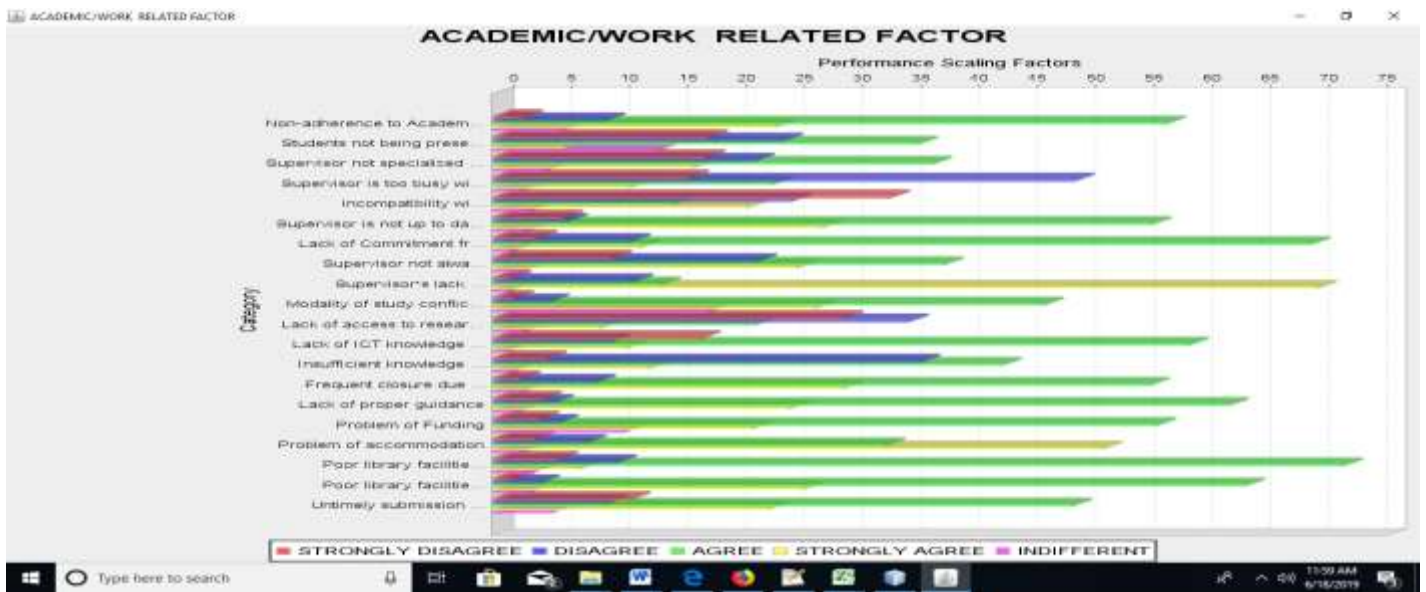


Figure 4.37: Analysis on Academic Factors

As shown in Figure 4.38, factors like regular hang out with friends/family, regular use of internet and social media and encouragement from sponsors/partners were attributing factors to student's performance had high scalability.

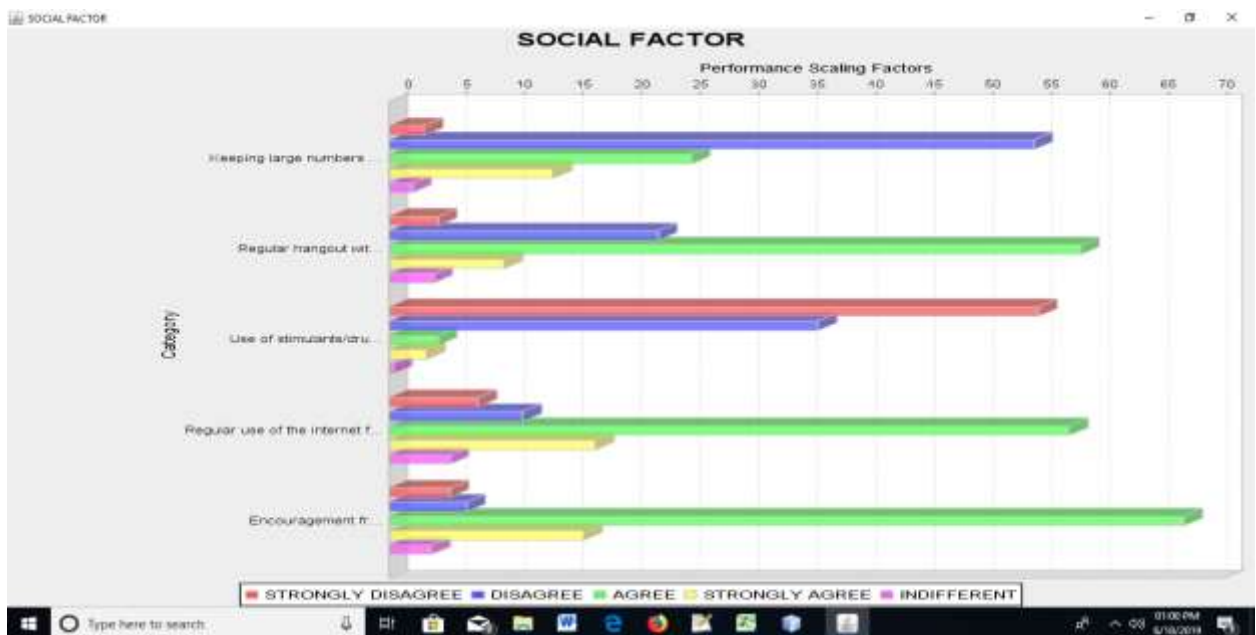


Figure 4.38: Analysis on Social factors

#### 4.7.3.4.4 Naïve Bayes Classification

Figure 4.39 shows the Pie Chart Interface for Naïve Bayes Classification. From the Chart, the accuracy of Naïve Bayes is 81.7%, correctly classified instances is 90.982%, and incorrectly classified instances is 9.018 respectively.

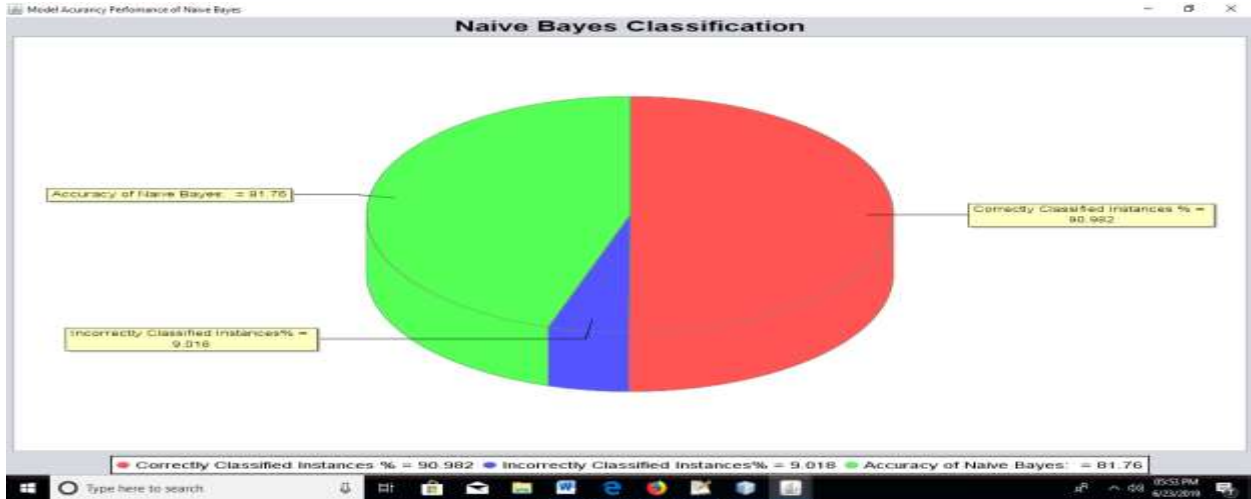


Figure 4.39 Pie Chart for Naïve Bayes Classification

Figure 4.40 shows the Bar Chart Interface for Naïve Bayes Classifier. The result was classified into HIGH, LOW and MEDIUM performance and was used to evaluate the student. The high bar represent student with highest grade, low bar represent student with lowest grade, same as the class of medium grade too.

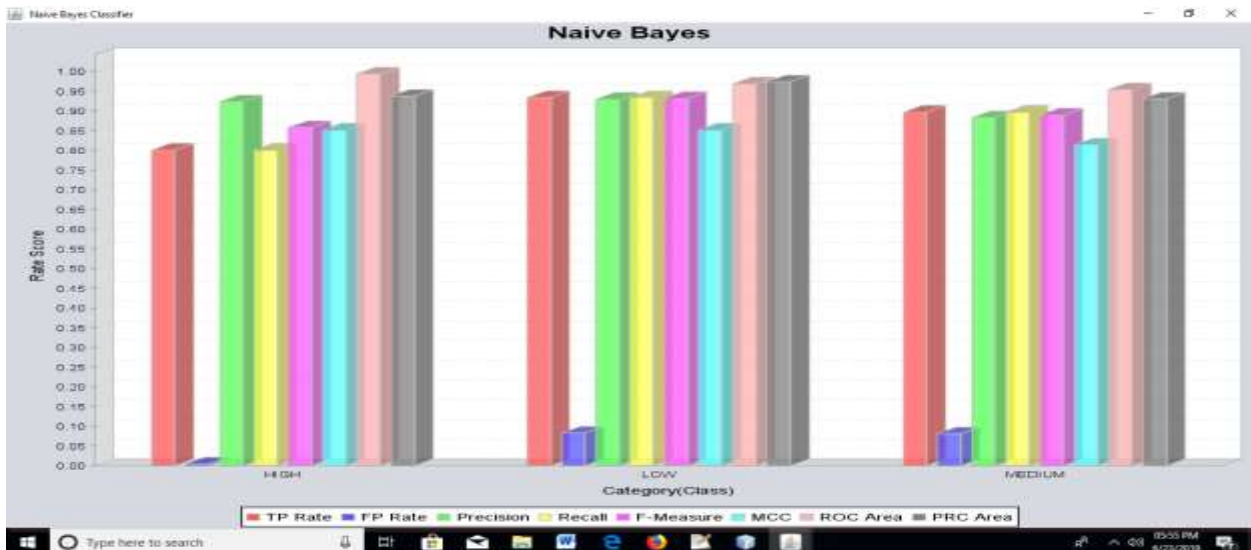


Figure 4.40 Model Building and Evaluation bar chat Interface for Naïve Bayes

Figure 4.41 shows the Pie Chart Interface for K-NN Classification. From the Chart, the accuracy of k-nearest neighbor is 51.91%, correctly classified instances is 60.321%, and incorrectly classified instances is 39.679%.

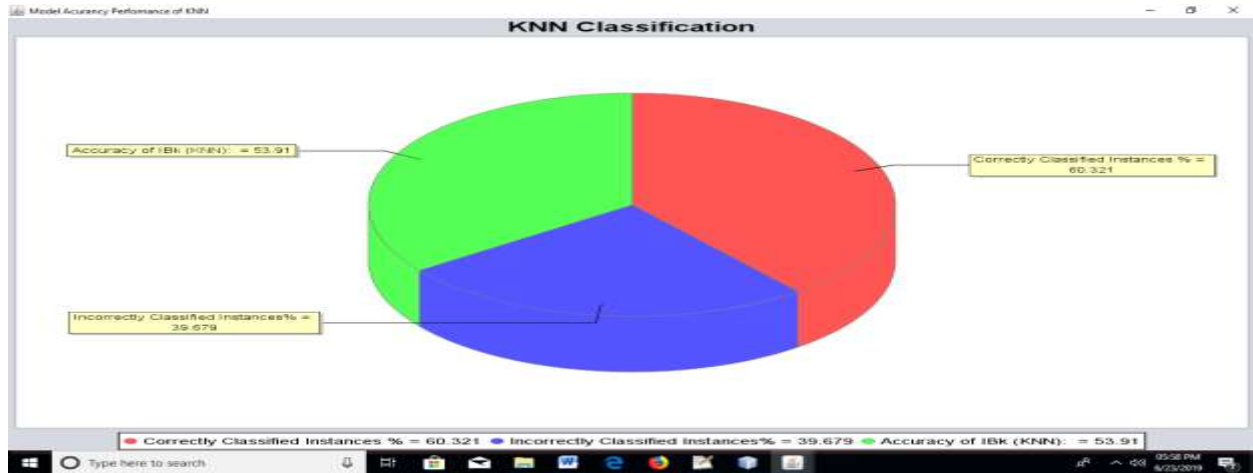


Figure 4.41 Pie Chart Interface for K-NN Classification

Figure 4.42 shows the Bar Chart Interface for K Nearest Neighbour Classifier. The result was classified into HIGH, LOW and MEDIUM performance and was used to evaluate the student. The high bar represent student with highest grade, low bar represent student with lowest grade, same as the class of medium grade too.

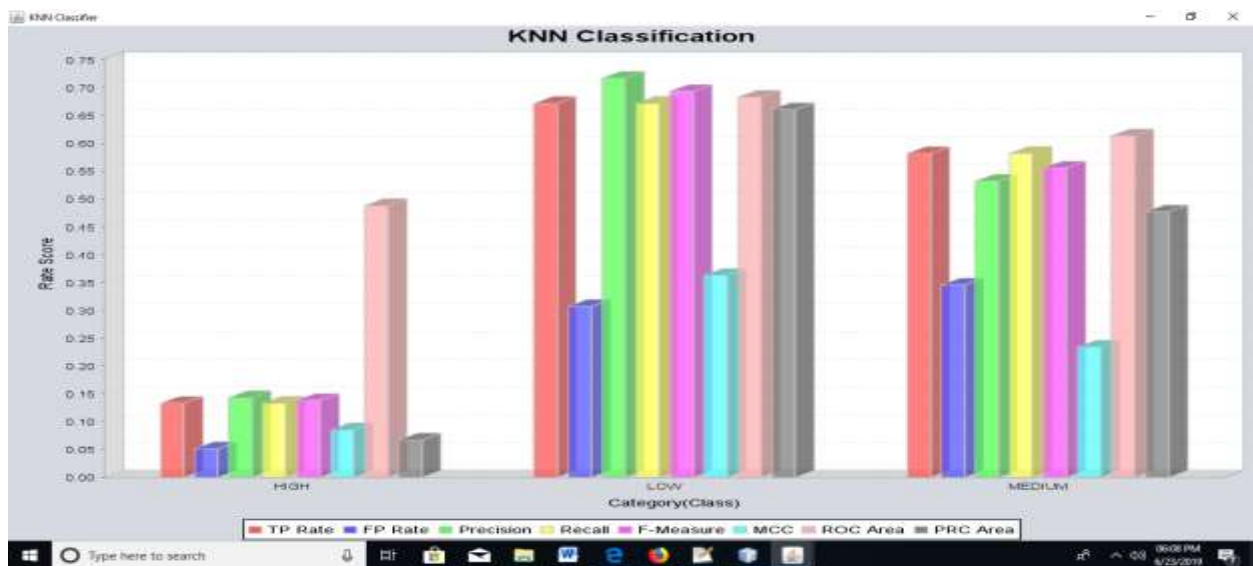


Figure 4.42 Model Building and Evaluation bar chat Interface for K-NN

Figure 4.31 and 4.32 shows the Receiver Operator Characteristics (ROC) for False Positive Rate (FPR) vs. True Positive Rate (TPR) for naïve bayes and KNN respectively

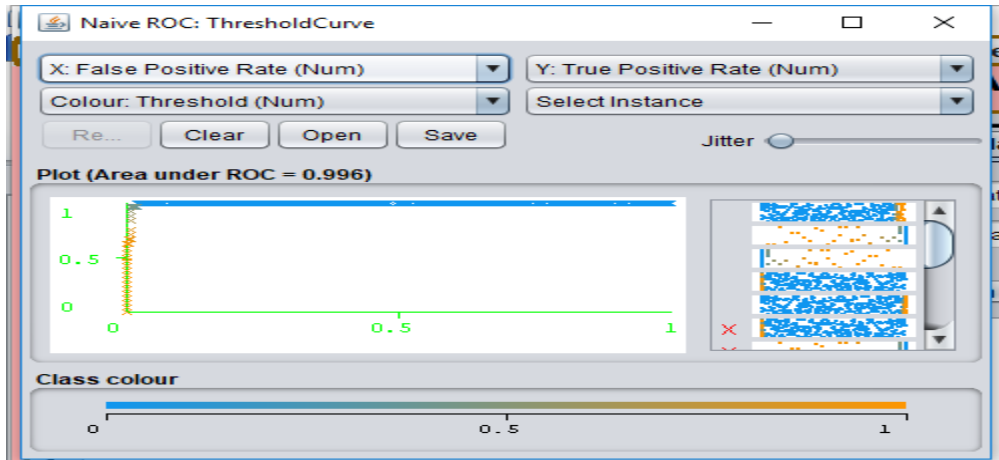


Figure 4.43 Receiver Operator Characteristics (ROC) for Naive Bayes

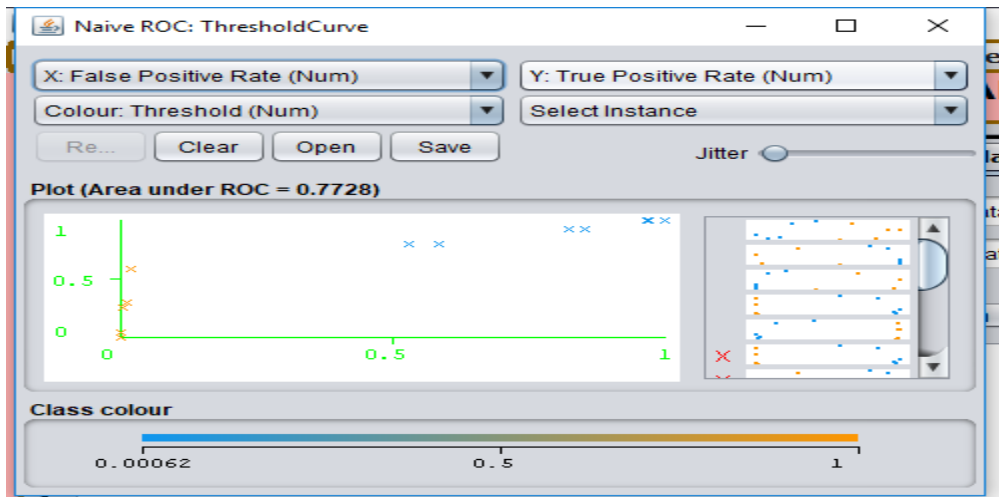


Figure 4.44 Receiver Operator Characteristics (ROC) for K-NN

Figure 4.45 and 4.46 represents the bar and pie chart for the two classifiers with their performance metrics. The various metrics are represented with various colours to clearly show the classification was done.

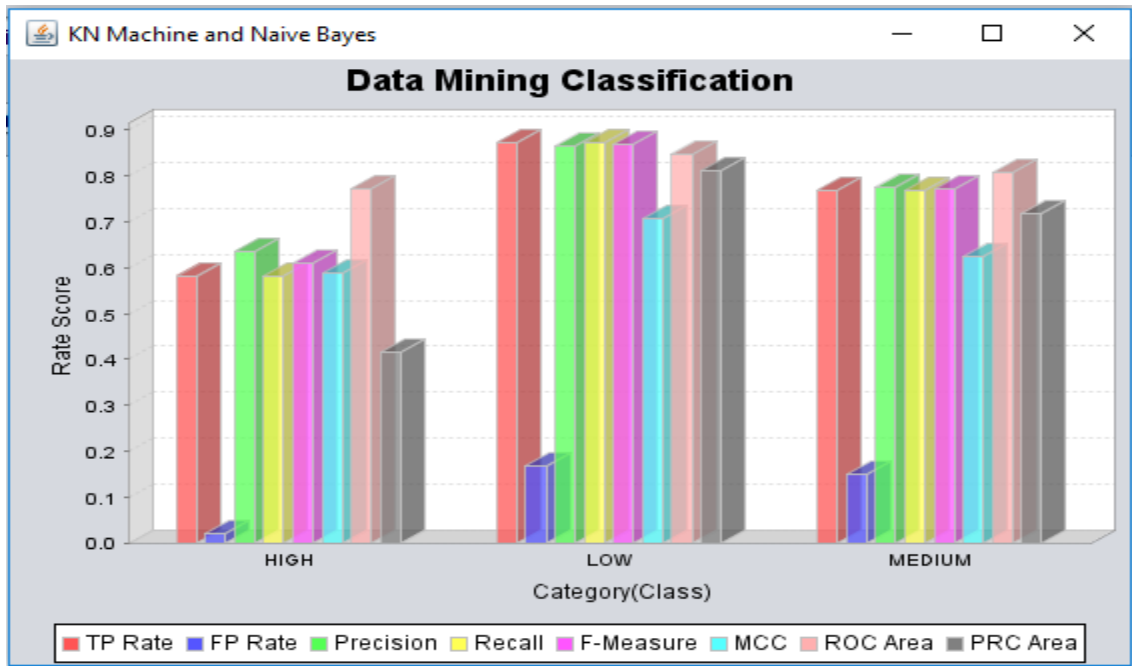


Figure 4.45 K-NN and Naïve Bayes bar chart showing their performance metrics

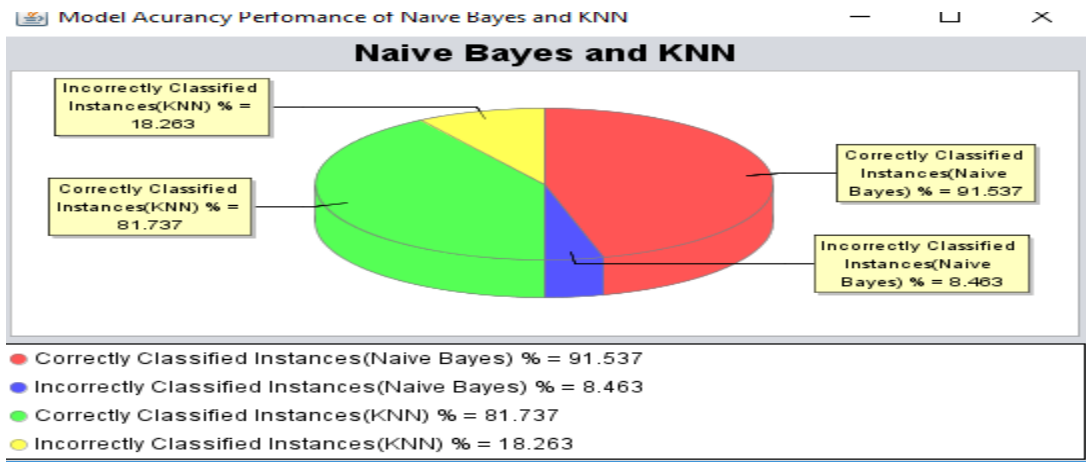


Figure 4.46 Naïve Bayes and KNN pie chart showing the classification accuracy

The chart in figure 4.47 describes the three levels of categories of model performance evaluation. K-Bay has the highest accuracy followed by Naïve and lastly k-nearest neighbor. The proposed system for student performance prediction was evaluated based on this criteria and the outcome were achieved.



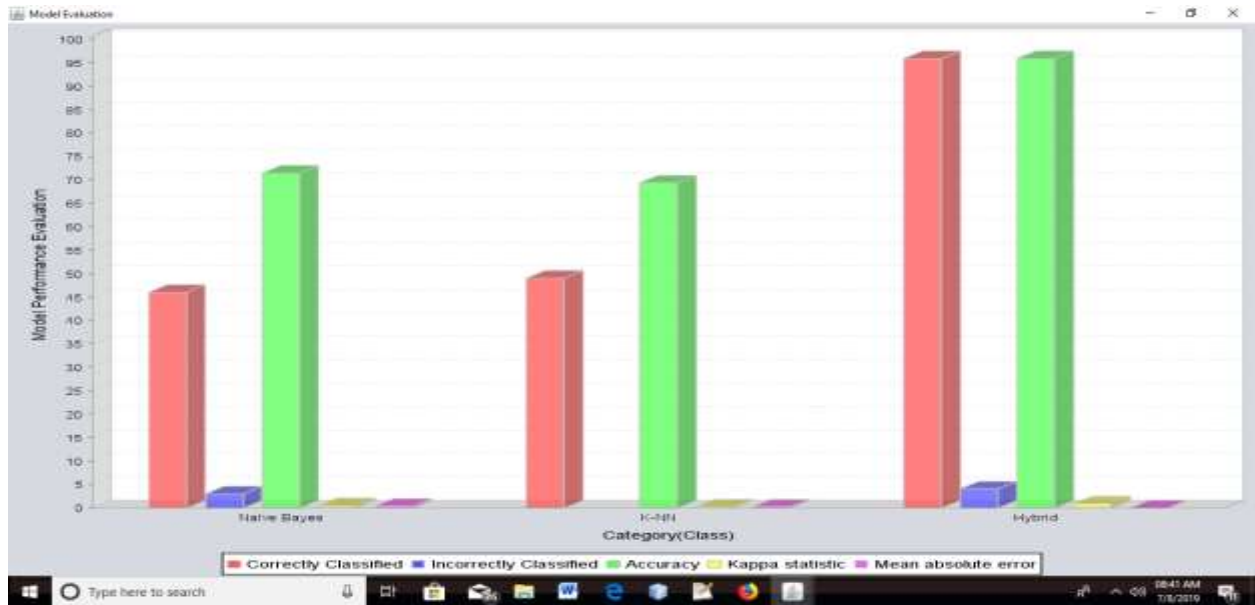


Figure 4.47 Bar Chart for Performance Evaluation of the K-Bay model

#### 4.7.3.5 Limitations of the System

1. One of the limitations of the new system is that it is limited to structured data set.
2. Only two classification algorithms were used

#### 4.7.4 System Security

The security of the system is an important factor as computing systems become more essential to our daily lives. It becomes ever more important that the services they provide are available whenever one needs them. The security installed in the new system includes the use of username and password to prevent unauthorized use of the system and access. There is a provision for Backup and restore in case of loss of information or system failure.

##### 4.7.4.1 Password Protection

Password protection was done in order to prevent unauthorized user hacking into system without the notice of administrator. The administrator needs to grant access to the user before he/she is permitted to use the services or application.

##### 4.7.4.2 Authentication

The authentication was done for both admin and user. It requires both username and password created during the execution program development phase. The username and password are stored at the back end (Database) and at the front end, the user is expected to input his/her username and password correctly before he/she is granted access to the application

#### **4.7.5 Training**

Training is very important aspect of system implementation. It enables the users operate the new system correctly and enjoy its benefits. The personnel in the system must know in detail what their roles will be, how they can use the system, and what the system will or will not do. The success or failure of well-designed and technically elegant systems depends on the way it is operated and used.

It is recommended that users of the new system are trained on its functionality and the parameters needed so as to enable them make maximum use of the new system. Handbooks, journals and lectures may be used as aids in training of staff.

#### **4.7.6 Documentation**

Documentation is critical to an effective program implementation. It is a written text that accompanies the new system that was developed. It explains how the system operates or how to use it and explains the different roles of different individuals. The documentation in the work focuses on how the application will be installed and used. To install it on the system to run from the hard disk, follow the procedure below.

- i. Install Java jdk version 1.7.0 on the Computer
- ii. Install Weka Machine learning version 3.8.0
- iii. Install Jade Agent platform
- iv. Install Java Virtual Machine
- v. Install Netbeans IDE version 7.3.0
- vi. Install nepad++

- vii. Install MySQL Sever (Wamp)
- viii. Insert CD-ROM into the system
- ix. Click Drive:
- x. Select the folder “MultiAgentPlatform”
- xi. Paste in the already installed Netbeans Project
- xii. Launch the Netbeans project to have access to application
- xiii. In the MultiAgentPlatform upload your database to MySQL server:
- xiv Right click on the application you just uploaded int Netbeans Project ” MultiAgentPlatform”
- xv. Click Run
- xvi. Enter the user name and password and click login
- xvii. Select options from the menu for the operation of the system

#### **4.7.7 SYSTEM CONVERSION**

It is a process of migrating from the old system to the new one. It provides understandable and structured approach to improve the communication between management and project team.

- a. **Parallel Conversion:**In a parallel changeover, the new system runs simultaneously with the old for a given period of time. Of all the techniques, this tends to be the most popular, mainly because it carries the lowest risk. If something goes wrong at any point, the entire system can be reverted back to its original state. A primary disadvantage in running two systems at the same time is higher costs. The parallel changeover process also can be quite time-consuming.
- b. **Pilot conversion:** this is quite similar to parallel implementation or conversion, however, with the pilot system only a portion of the new system is run alongside the new system. Since parallel changeovers tend to be expensive, using the pilot changeover technique allows companies to run the new system next to their old but on a much smaller scale. This makes the pilot changeover method much more cost-effective.

- c. Direct conversion:referred to as immediate replacement, it tends to be the least favourite of the changeover techniques. In a direct changeover, the entire system is replaced in an instant. Basically, as soon as the new system is powered up, the old system is shut down. Advantages of Direct Changeover are that it forces users to make new system work and it offers immediate benefit from new methods and control. This type of changeover carries the most risk because; if something goes wrong, reverting back to the old system usually is impossible. Using the direct changeover technique tends to work best in situations where a system failure isn't critical enough to result in a disaster for the company or where there are no exiting systems..
- d. Phase conversion: Phased changeover technique is considered a compromise between parallel and direct changeovers. In a phased changeover, the new system is implemented one stage at a time. As an example, consider a company working toward installing a new financial system. Implementing the new system one department at a time, the company converts accounts receivable, accounts payable, payroll, and so on. Advantages to phased changeovers are their low cost and isolated errors. The main disadvantage is the process takes a long time to complete because phases need to be implemented separately.

#### **4.7.7.2 Recommended Procedure**

Direct Change over is recommended for this system since there are no existing system for performance prediction.

## CHAPTER FIVE

### SUMMARY, CONCLUSION AND RECOMMENDATION

#### 5.1 Summary

Educational Data Mining system remains relevant in analyzing and predicting the Academic Performance of students by considering different performance factors.

Artificial intelligence techniques have been successfully used in many fields and it offers potential possibility of classifying students' performance. Predicting students' performance is an active research area due to its significant importance to staff, students and academic institution at large. It helps educators and learners improve their learning and teaching process. The knowledge will improve the quality of education, performance of students and help decrease failure rate.

In this research paper, we proposed K-Bay, a hybrid classifier that combines Naïve Bayes and K-Nearest Neighbor in order to predict students' success and failure rates from a collected training dataset. First, a survey was constructed that targeted Postgraduate students where information on their personal, social, and academic data related to them were collected.

Secondly, the collected dataset was preprocessed and explored to become appropriate for the data mining tasks. Naïve Bayes and K- Nearest Neighbour (KNN) Algorithm were evaluated in terms of their operations using WEKA free software tool on the data. Feature selection technique was used in ranking the given attributes/ features to identify the most important features and their order of ranking. Using correlation based feature selection; five attributes were ranked most important; they include Employment-Status, Supervisor being too busy with extensive commitment, Poor library facilities, standard equipment and Laboratory, Use of stimulants, drugs to enhance study, regular-use-of-the-internet-for-surfing-and-social-networking

Thirdly, a hybrid model was developed using a combination of Naïve Bayes and K-Nearest Neighbor for predicting students' academic performance. First semester results Result of 499 postgraduate students were used in the prediction.

Fourthly, the new model was tested in two parts; using all the attributes and using highly influencing attributes. Using all the attributes, the system realized an accuracy of 95.92% as against the single classifiers; Naïve Bayes and KNN which had an accuracy of 69.39% and

71.43% respectively; Execution time for the new model was 0.134 seconds while KNN and Naive Bay was 0.357 and 0.18 seconds respectively. Using only the highly influencing attributes, the system realized an accuracy of 99% as against the single classifiers; Naïve Bayes and KNN which had an accuracy of 75.51% and 59.18% respectively.

In conclusion, this study can motivate and help universities to perform data mining tasks on their students' data regularly to find out interesting results and patterns which can help both the university as well as the students in many ways.

## **5.2 Conclusion**

The research has revealed that data mining has a potential to become a serious part of educational institutions' decision making and knowledge management process, The research shows that students' data that are available to higher educational institutes can be used to predict the academic performance of students enrolled for various courses using educational data mining techniques. Academic performance of student isn't a result of only one deciding factor. It hinges on various factors like personal, socio-economic, psychological and other environmental factors.

Predicting students' academic performance with a high rate of accuracy enhances educational services at the higher institution. The reliability of the model can help the institution to know the academic status of students in advance and identify students who have higher chances of failing so that appropriate action can be taken such as advising students and providing remedy on time. It also enables institution to identify bright students and nurture their future growth by encouraging them. In the long run, it can help the student improve in their academics and eventually leads to a better performance thereby reducing drop out and depression rates on the part of the students.

## **5.3 Recommendation**

A model for predicting students' academic performance using K-Bay, a hybrid of K-Nearest Neighbor and Naïve Bayes Classification has an improved accuracy and can easily be implemented in institutions of higher learning to do prediction of students' performance and also mine interesting features pertaining academics of students.

It is recommended that all educational institutes adopt the model in this dissertation to help predict students' performance on time, offer academic advice to the students to failing students and help improve students' performance in the forth coming semesters.

Researchers should conduct periodic cross-validation studies and update the model as needed to ensure that the model accuracy is not compromised due to changes in student cohorts, course offerings, and instructional assessments.

### **5.3.1 Application Areas**

Curriculum committees can use prediction results to guide changes to the curriculum and evaluate the effects of those changes. An academic advisor can refer to the prediction results when giving advice to students who perform weakly in their studies so that preventive measures can be taken much earlier. In addition, an instructor can further improve his/her teaching and learning approach, as well as plan interventions and support services for weak students.

### **5.3.2 Suggestion for Further Research**

1. More advanced data mining technique should be considered for the classification of students' performance to acquire a wider approach and more reliable outputs.
2. More student and ample input data can further be used to produce accurate results. With more and more demand for not only student but also performance prediction as a whole, there is a lot of data that can be taken into consideration for more accurate results
3. The scope should be expanded to include more Department and Programmes to give different ideas and allow the university to gain better understanding for academic performance of the students

## **5.4 Contribution to Knowledge**

The work focused on developing a hybrid model, K-Bay (combination of KNN and Naïve Bayes algorithm) to classify the various expected performance levels (high performance, low performance and medium performance)for postgraduate students.

The second contribution of this study is being able to extract the attribute importance ranking of attributes to determine which factors have significant contribution to the prediction of the overall academic performance.

The third contribution is to provide a reference and comparative study for the next researcher in application of the K-Bay prediction system



## REFERENCES

- Acharya, A., & Sinha, D. (2014). Early prediction of student performance using machine learning techniques. *International Journal of Computer Applications*, Volume 107–No. 1, December 2014.
- Ahmad, F., Ismail, N., & Aziz, A. (2015). The prediction of students' academic performance using classification data mining techniques. 9. 6415-6426. 10.12988/ams.2015.53289.
- Al-Barky. A., & Ali, J. (2012). Intelligent mining agent. 8th International Conference on Computing Technology and Information Management (ICCM), 2012, vol. 1, p. 23.
- Albashiri K., (2010) An investigation into the issues of multi-agent data mining, Ph.D, University of Liverpool, 2010
- Alcala-Fdez, J., Del-Jesus, M., Ventura, S., Garrell, J., Otero, J., Romero, C., Bacardit, J., Rivas, V., Fernandez, J., & Herrera, F., (2009), : “KEEL: A software tool to Assess Evolutionary Algorithms to Data mining Problems”, *Soft computing* 13:3, pp 307-318(2009).
- Algarni, A. (2016). Data Mining in Education. *International Journal of Advanced Computer Science and Applications*.7. 10.14569/IJACSA.2016.070659.
- Allen, J., Smith, C., & Muehleck, J. (2014). Pre-and post-transfer academic advising: What students say are the similarities and differences. *Journal of College Student Development*, 55(4), 353-367.
- Altujjar, Y., Altamimi, W., Al-Turaiki, I., & Al-Razgan, M.(2016), Predicting Critical Courses Affecting Students Performance: A Case Study. *Procedia Computer Science* 82 ( 2016 ) 65 71.
- Amin. Z., Refik, C., Yau, H., & Hernandez-Torrano, D., (2017). Predicting Students“ GPA and Developing Intervention Strategies Based on Self-Regulatory Learning Behaviors. 2017, IEEE
- Amra, I., Abu, M., & Ashraf Y. (2017). Students Performance Prediction Using KNN and Naïve Bayesian. 2017 8th International Conference on Information Technology (ICIT)
- Amrieh, E., Hamtini, T., & Aljarah, I., (2016). Mining educational data to predict student's academic performance using ensemble methods. *International Journal of Database Theory and Application* 9(8), 119-136.
- Arlot, S., & Celisse, A., (2010). A survey of cross-validation procedures for model selection. *Statistics Surveys*, vol. 4, 2010, pp. 40-79

- Arockiam, L., Charles, S., Carol, I., Bastin, P., Thiagaraj, S., & Yosuva, V., (2010). Deriving Association between Urban and Rural Students Programming Skills. *International Journal on Computer Science and Engineering* Vol. 02, No. 03, 2010, 687-690
- Arsad,P., Buniyamin, N., & Manan, J., (2013). A Neural Network Students' Performance Prediction Model (NNSPPM). IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA), 26-27 November 2013.
- Asiah, M., Khidzir, Z., Safaai, D., Hafzan, N., Yaacob, M., Mohd S., & Siti S., (2019). A Review on Predictive Modeling Technique for Student Academic Performance Monitoring. MATEC Web of Conferences 255, 03004 (2019) <https://doi.org/10.1051/mateconf/201925503004> EAAI Conference 2018
- Asif, R., Merceron, A., & Pathan, M. (2015a). Investigating performance of students: A longitudinal study. In 5th international conference on learning analytics and knowledge (pp. 108e112). Poughkeepsie, NY, USA, March 16-20 <http://dx.doi.org/10.1145/2723576.2723579>.
- Asif, R., Merceron, A., & Pathan, M. (2015b). Predicting student academic performance at degree level: A case study. *International Journal of Intelligent Systems and Applications (IJISA)*, 7(1), 49e61. <http://dx.doi.org/10.5815/ijisa.2015.01.05>.
- Avula, A., & Arba, A., (2018). Improving Prediction Accuracy Using Hybrid Machine Learning Algorithm on Medical Datasets. *International Journal of Scientific & Engineering Research* Volume 9, Issue 10, October-2018 1461 ISSN 2229-5518
- Ashraf, A., Sajid, A., & Muhammad, G. (2018). A Comparative Study of Predicting Student's Performance by use of Data Mining Techniques. *American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)* ISSN (Print) 2313-4410, ISSN (Online) 2313-4402
- Aziz, A., Ismail, N., & Fadhilah, A. (2014), First Semester Computer Science Students' Academic Performances Analysis by Using Data Mining Classification Algorithms. *Proceeding of the International Conference on Artificial Intelligence and Computer Science(AICS 2014)*, September 2014.
- Bhardwaj, B., & Pal, S. (2011). Data Mining: A prediction for performance improvement using classification *International Journal of Computer Science and Information Security (IJCSIS)*, vol 9, iss 4, 2011
- Badr, G., Algobail, A., Almutairi, H., & Almutery, M. (2016). Predicting Students Performance in University Courses: A Case Study and Tool in KSU Mathematics Department. *Procedia Computer Science* 82 ( 2016 ) 80 89.

- Bakar. A., Othman, Z., Hamdan, A., Yusof, R., & Ismail, R. (2008). Agent based data classification approach for data mining. In International Symposium on Information Technology, vol 2, pp. 1--6, 2008.
- Baker. R., & Inventado, P. (2014). Educational data mining and learning analytics. Learning Analytics: From Research to Practice, Springer New York, 2014, pp. 61–75.
- Baker, R. (2014). Educational Data Mining: An Advance for Intelligent Systems in Education. AI Educ., pp. 78–82, 2014.
- Bansode, J. (2016). Mining Educational Data to Predict Student's Academic Performance. *International Journal on Recent and Innovation Trends in Computing and Communication* ISSN: 2321-8169, Volume: 4 Issue: 1, 2016.
- Baradwaj, B., Pal, S., (2012). Mining educational data to analyze students' performance. IJACSA 2: 63-69.
- Baradwaj, B., & Kumar, B., (2011). Mining Educational Data to Analyze Students Performance. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 2(6), 2011, 63-69
- Bhardwaj, K., Brijesh, P., & Pal, S. (2012). Data Mining: A prediction for performance improvement using classification. *International Journal of Computer Science and Information Security*.
- Bhullar. M., & Kaur, A., (2012) Use of Data Mining in Education Sector. Proceedings of the World Congress on Engineering and Computer Science (WCECS), San Francisco, USA, October 2012.
- Bigus,J., (1996). Data Mining with Neural Networks - Solving Business Problems. Application Development to Decision Support, McGraw-Hill, 1996.
- Bilal, S., (2017), A Comparative Analysis of Exam Timetable Using Data Mining Techniques.. *IJCSNS International Journal of Computer Science and Network Security*, VOL.17 No.1, January 2017
- Blagojević, M., & Micić, Ž., (2013). A web-based intelligent report e-learning system using data mining techniques. *Computers & Electrical Engineering*. 39. 465–474. 10.1016/j.compeleceng.2012.09.011.
- Bratu, C., Muresan, T., & Potolea, R. (2008). "Improving classification accuracy through feature selection". In Intelligent Computer Communication and Processing, 2008.ICCP 2008.4th International Conference on, pages 25{32.IEEE.

- Brijesh K., & Saurabh P., (2011), "Data Mining: A prediction for performance improvement using classification. *International Journal of Computer Science and Information Security*, Vol. 9, No. 4, April 2011, pp 136-140.
- Bringsjord, S., & Govindarajulu, N (2018), "Artificial Intelligence", *The Stanford Encyclopedia of Philosophy* (Fall 2018 Edition), Edward N. Zalta (ed.), URL = <<https://plato.stanford.edu/archives/fall2018/entries/artificial-intelligence/>>. Aug 17, 2018
- Chang L., (2008)"Applying Data Mining to Predict College Admissions Yield", Spring 2008 - College of Education, <http://www.ed.psu.edu/educ/eps/ir-certificate/downloadsforms/hied-860-syllabus-su11.doc>.
- Cortez P, & Silva A., (2008) "Using Data Mining To Predict Secondary School Student Performance", In EUROSIS, A. Brito and J. Teixeira (Eds.), 2008, pp.5-12.
- Delali K & Gyimah E (2017): "Students Grades Predictor using Naïve Bayes Classifier – A Case Study of University of Education", *Winneba International Journal of Innovative Research in Science, Engineering and Technology (A High Impact Factor, Monthly, Peer Reviewed Journal)* : [www.ijirset.com](http://www.ijirset.com) Vol. 6, Issue 10, October 2017. ISSN(Online): 2319-8753 ISSN (Print): 2347-6710
- Devasia T, Vinushree T., & Hegde V., (2016): "Prediction of Students Performance using Educational Data Mining", 2016, IEEE
- Dewan M., Zhang L., Rahman C., Hossain A., & Strachan R., (2014). Hybrid decision tree and Naive bayes classifiers for multi-class classification tasks. Elsevier, *Expert systems with applications*, pp) 1937-1946, 2014
- Divyabharathi, Y., & Someswari, P. (2018), A framework for student academic performance using naive Bayes classification technique. *Journal of Advancement in Engineering and Technology* 6(3):1–4, 2018
- Donaldson, P., McKinney, L., Lee, M. & Pino, D. (2016). First-year community college students' perceptions of and attitudes toward intrusive academic advising. *NACADA Journal* 36(1), 30-42
- Gayathri, E., Aarthi, N., (2014), Application of Data Mining in Educational Database for Predicting Behavioural Patterns of the Students. *International Journal of Computer Science and Information Technologies(IJCSIT)*, Vol. 5 (3) , 2014, 4649-4652
- Erlich, R., & Russ-Eft, D., (2013). Assessing student learning in academic advising using social cognitive theory. *NACADA Journal*, 33(1), 16–33.

- Etzold, D. (2003). Improving spam filtering by combining Naïve Bayes with simple K-Nearest Neighbor searches. arXiv:cs/0312004v1 [cs.LG].
- Fadhila, A., Ismail, N., & Azwa, A. (2015). The Prediction of Students' Academic Performance Using Classification Data Mining Techniques. *Applied Mathematical Sciences*, Vol. 9, 2015, no. 129, pp. 6415 – 6426. <http://dx.doi.org/10.12988/ams.2015.53289>
- Fangming G., & Hua S (2010), "Research and Application of Data-Mining Technique in Timetable Scheduling" in Proceedings of Computer Engineering and Technology (ICCET), 2nd International Conference
- Fariz, A., Abouchabaka, J., & Rafalia, N. (2015). Using Multi-Agents Systems in Distributed Data Mining: A Survey. *Journal of Theoretical and Applied Information Technology*, vol. 73 No. 3, pp. 427-440, March 2015.
- Fayyad, U., Piatetsky-Shapiro, P., & Uthurusamy, F. (1996). *Advances in Knowledge Discovery and Data Mining*, (AKDDM). AAAI/MIT Press.
- Febrianti, W., & Tjhin, V. (2017). Predicting Students Performance in Final Examination using Linear Regression and Multilayer Perceptron. 978-1-5090-4688-1/17/\$31.00 ©2017IEEE
- García-Saiz, D., & Zorrilla, M. (2011). Comparing classification methods for predicting distance students' performance. Proceedings of the Second Workshop on Applications of Pattern Analysis, 26-32
- Garima, S., Santosh, K. (2017). Analysis and Prediction of Student's Academic Performance in University Courses. *International Journal of Computer Applications* (0975 – 8887) Volume 160 – No 4, February 2017
- Ghahramani, Z., (2004). Unsupervised learning," in advanced lectures on machine learning, Ed: Springer, 2004; pp. 72-112. [https://doi.org/10.1007/978-3-540-28650-9\\_5](https://doi.org/10.1007/978-3-540-28650-9_5)
- Golding, P., & Donaldson, O. (2006) "Predicting Academic Performance. Paper presented at the frontiers in Education Conference, 36th Annual, 27-31.
- Gray, S., & Rogers, M., Martinussen, R., & Tannock, R. (2014). Longitudinal relations among inattention, working memory, and academic achievement: Testing mediation and the moderating role of gender. *PeerJ*. 3. e939. [10.7717/peerj.939](https://doi.org/10.7717/peerj.939).
- Guruler, H., Istanbulu, A., & Karahasan, M. (2010). A new student performance analysing system using knowledge discovery in higher educational databases. *Computer and Education*. 2010. 247-254.

- Habley, W. (2003). *Faculty advising: Practice and promise*. Boston, MA: Anker: Faculty advising examined: Enhancing the potential of college faculty as advisors.
- Habley, W., (2004). *The status of academic advising: Findings from the act sixth national survey* (monograph no. 10). Manhattan, KS: National Academic Advising Association
- Hämäläinen, W., & Vinni, M. (2010). Classifiers for educational technology. In C. Romero, S. Ventura, M. Pechenizkiy, R.S.J.d. Baker (eds.), *Handbook of Educational Data Mining*, (pp. 54- 74). CRC Press.
- Han, E., Karypis G., & Kumar, V. (1999). Text categorization using weight adjusted k-nearest neighbour classification. Technical Report, Department of Computer Science and Engineering, Army HPC Research Center, University of Minnesota. <http://glaros.dtc.umn.edu/gkhome/fetch/papers/wknnPAKDD01.pdf>.
- Han, J., & Kamber, M. (2000). *Data Mining: Concepts and Techniques*. Morgan Kaufmann, San Francisco.
- Han,J., & Kamber, M., (2006). *Data Mining Concepts and Techniques*. 2nd ed. San Francisco: Morgan Kaufmann, pp.5-7, 2006.
- Hemamalini, R., & Josephine, M., (2014). An Analysis on Multi-Agent Based Distributed Data Mining System. *International Journal of Scientific and Research Publications*, Volume 4, Issue 6, June 2014 ISSN 2250-3153
- Hershkovitz, A., & Nachmias, R., (2008), Developing a log-based motivation measuring tool. EDM, pp. 226–233, Citeseer, 2008.
- Hilal, A. (2017). Analysis of Students' Performance by Using Different Data Mining Classifiers. IJ. Modern Education and Computer Science, 2017, 8, 9-15 Published Online August 2017 in MECS (<http://www.mecs-press.org/>) DOI: 10.5815/ijmecs.2017.08.02
- Hong, S., & Weiss, S. (2012). *Advances in Predictive Model Generation for Data Mining*. IBM Research Report RC-21570
- Hsiao, H., Chen, S., Chang, J., & Tsai, P., (2008). Predicting Sub cellular Locations of Eukaryotic Proteins Using Bayesian and K-Nearest Neighbor Classifiers. *Journal of Information Science and Engineering*, 24(5), 1361-1375.
- Hunter, M., & White, E. (2004). Could fixing academic advising fix higher education? *About Campus*, 9(1), 20-25.

- Jadhav, S., & Channe, H. (2017). Comparative Study of K-NN , Naive Bayes and Decision Tree Classification Techniques. vol. 5, no. 1, pp. 2014–2017, 2016.
- Jang, J. (1993). Adaptive-network-based fuzzy inference system. *Systems, Man and Cybernetics. IEEE Transactions on*, 1993; 23: 665-685.<https://doi.org/10.1109/21.256541>
- Jiang, L., Wang, D., Cai, Z., Jiang, S., & Yan, X. (2009). Scaling Up the Accuracy of K-Nearest-Neighbour Classifiers: A Naive-Bayes Hybrid. *International Journal of Computers and Applications*, 31(1). <http://dx.doi.org/10.2316/Journal.202.2009.1.202-2453>
- Jiang, L., Zhang, H., & Cai, Z. (2006). Dynamic K-Nearest-Neighbor Naive Bayes with Attribute Weighted. *Proceedings of the 3rd International Conference on Fuzzy Systems and Knowledge Discovery, LNAI 4223*, pp.365-368.Springer Press. [http://dx.doi.org/10.1007/11881599\\_41](http://dx.doi.org/10.1007/11881599_41)
- Jindal, R., & Borah, M. (2013). A Survey on Educational Data Mining and Research trends. *International Journal of Database Management System (IJDMS)*, 5(3), 2013, 53–73.
- Johnson, S., Aragon, S., Shaik, N., & Palma-Rivas, N. (2000). Comparative analysis of learner satisfaction and learning outcomes in online and face-to-face learning environments. *Journal of interactive learning research*, 11(1):29.
- Kabakchieva, D., (2013). Predicting Student Performance by Using Data Mining Methods for Classification. *Cybernetics and information technologies Volume 13, No 1 Sofia • 2013* Print ISSN: 1311-9702; Online ISSN: 1314-4081 DOI: 10.2478/cait-2013-0006
- Kabra, R., & Bichkar, R., (2011). Performance Prediction of Engineering Students using Decision Trees. *International Journal of Computer Applications* 36(11):8-12, December 2011.
- Kalles, D., & Pierrakeas, C. (2006). Analyzing student performance in distance learning with genetic algorithms and decision trees. *Applied Artificial Intelligence* 20(8), 655-674.
- Kantardzic, M., (2003). *Data mining: Concepts, models, methods, and algorithms*. John Wiley & Sons, 2003.
- Kaur, P., Singh, M., & Josan, G. (2015). Classification and prediction based data mining algorithms to predict slow learners in education sector. *3rd International Conference on Recent Trends in Computing 2015(ICRTC-2015)*.
- Kaur, P., Manpreet, S., & Gurpreet, S. (2015). Classification and prediction based data mining algorithms to predict slow learners in education sector. *3rd International Conference on Recent Trends in Computing 2015(ICRTC-2015)*. *Procedia Computer Science* 57 (2015 ) 500 – 508

- Kay, J., Maisonneuve, N., Yacef, K., & Reimann, P. (2006). The big five and visualizations of team work activity. *Intelligent tutoring systems*, pp. 197–206, Springer, 2006.
- Keup, J., & Stolzenberg, E., (2004). The 2003 your first college year (yfcy) survey: Exploring the academic and personal experiences of first-year students. (monograph No. 40). Columbia, SC: University of South Carolina National Resource Center for The Freshman Year Experience and Students in Transition.
- Kesavaraj, G., & Sukumaran, S., (2013). A study on classification techniques in data mining. In *Computing, Communications and Networking Technologies (ICCCNT)*, 2013 Fourth International Conference on, 2013; pp. 1-7.
- Kolo, D., Solomon, A., & Alhassan, J., (2015). A Decision Tree Approach for Predicting Students Academic Performance. *I.J. Education and Management Engineering*, 2015, 5, 12-19 Published Online October 2015 in MECS (<http://www.mecspress.net>) DOI: 10.5815/ijeme.2015.05.02.
- Kotsiantis, S., Pierrakeas, C., & Pintelas, P. (2004). Prediction of Student's Performance in Distance Learning Using Machine Learning Techniques. *Applied Artificial Intelligence*, Vol. 18, No. 5, 2004, pp. 411-426
- Kotsiantis, S., Zaharakis, I., & Pintelas, P. (2007). *Supervised machine learning: A review of classification techniques*. Ed, 2007.
- Kotsiantis, S., Pierrakeas, C., & Pintelas, P. (2004). Prediction of Student's Performance in Distance Learning Using Machine Learning Techniques. *Applied Artificial Intelligence*, Vol.18,2004,No5,411-426.
- Koutina, M., & Kermanidis, K., (2011). Predicting Postgraduate Students' Performance Using Machine Learning Techniques. L. Iliadis et al. (Eds.): *EANN/AIAI 2011, Part II, IFIP AICT 364*, pp. 159–168, 2011. © IFIP International Federation for Information Processing 2011.
- Kovaicic, Z. (2002). Early Prediction of Student Success: Mining Students Enrolment Data. *Proceedings of Informing Science & IT Education Conference (InSITE'2010)*, 2010, 647-665. Luan, J. *Data Mining and Its Applications in Higher Education. – New Directions for Institutional Research, Special Issue Titled Knowledge Management: Building a Competitive Advantage in Higher Education*, Vol. 2002, 2002, Issue 113, 17-36.
- Kumar, S., & Vijayalakshmi, M. (2011). A Novel Approach in Data Mining Techniques for Educational Data. *3rd Int. Conf. Mach. Learn. Comput. (ICMLC 2011) A*, no.Icmlc, pp. 152–154, 2011.



- Kumar, M., & Singh, J., & Handa, D. (2017). Literature Survey on Student's Performance Prediction in Education using Data Mining Techniques. *International Journal of Education and Management Engineering*. 6. 40-49. 10.5815/ijeme.2017.06.05.
- Kuyoro, S., Goga, N., Awodele, O., Okolie, S. (2013). Optimal Algorithm for Predicting Students' Academic Performance. *International Journal of Computers & Technology*, Volume 4 No. 1, Jan-Feb, 2013 ISSN 2277-3061
- Lakshmi, T., Martin, A., Mumtaj, B., & Prasanna, V. (2013). An Analysis on Performance of Decision Tree Algorithms using Student's Qualitative Data. *I.J. Modern Education and Computer Science*, May 2013.
- Lam, H., Chin, H., Wan, T., & Hui, M. (2010). A Review of Nearest Neighbor-Support Vector Machines Hybrid Classification Models. *Journal of Applied Sciences*, 10: 1841-1858.
- Madhav, S., & Reshma, G. (2017). Predicting Student's Performance using CART approach in Data Science. *International Conference on Electronics, Communication and Aerospace Technology ICECA 2017*
- Makhtar, M., Nawang, H., & Shamsuddin, S. (2017). Analysis on students performance using Naïve Bayes classifier. *J. Theor. Appl. Inf. Technol.* 95(16):3993–3999, 2017.
- McCann, S., & Lowe, D. (2012). Local naive bayes nearest neighbor for image classification. *CVPR*.
- Mgala, M., & Mbogho, A. (2015). Data-driven intervention-level prediction modeling for academic performance. 15 *Proceedings of the Seventh International Conference on Information and Communication Technologies and Development*. May 2015. DOI: 10.1145/2737856.2738012
- Mierswa, I., Wurst, M., Klinkenberg, R., Scholz, M., & Euler, T. (2006). YALE: Rapid Prototyping for Complex Data Mining tasks. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD-06)*, pp. 935-940, 2006.
- Mikut, R., & Wiley, M. (2011). *Interdisciplinary Reviews: Data Mining and Knowledge Discovery*. Volume 1, Issue 5, pages 431–443, September/October 2011.
- Minaei-Bidgoli, B., Kortemeyer, G., & Punch, W. (2004). Enhancing Online Learning Performance: An Application of Data Mining Method. *7th IASTED International Conference on Computers and Advanced Technology in Education (CATE 2004)*, 2004.

- Minaei-Bidgoli, B., Kashy, D., Kortemeyer, G., & Punch, W. (2003). Predicting student performance: An application of data mining methods with an educational web-based system. Proceedings of the 33rd Annual Conference on Frontiers in Education, Nov. 5-8, IEEE Computer Society, Washington, DC, USA., pp: 13-18.
- Mishra, T., Kumar, D., & Gupta, S. (2014). Mining Students' Data for Prediction Performance. International Conference on Advanced Computing and Communication Technologies, ACCT. 255-262. 10.1109/ACCT.2014.105.
- Mohan, M., Siju, K., & Kumari, R. (2015). A Big Data Approach for Classification and Prediction of Student Result Using Map Reduce. 2015 IEEE Recent Advances in Intelligent Computational Systems (RAICS) | 10-12 December 2015
- Mahdi, H., & Mohamed, H., & Attia, S. (2011). Data Mining for Decision Making in Multi-Agent Systems. 10.5772/15584.
- Moucary, C., Khair, M., & Zakhem, W. (2011). Improving student's performance using data clustering and neural networks in foreign-language based higher education. The Research Bulletin of Jordan ACM, 2(3), pp 27-34
- Nakayama, M., Kouichi, M., Hiroh, Y. (2018). Contributions of Student's Assessment of Reflections on the Prediction of Learning Performance. 2018, IEEE
- Namdeo, J., & Jayakumar, N. (2014), Predicting Students Performance Using Data Mining Technique with Rough Set Theory Concepts. *International Journal of Advance Research in Computer Science and Management Studies* Volume 2, Issue 2, February 2014.
- Naren., J. (2014). Application of Data Mining in Educational Database for Predicting Behavioral Patterns of the Students. *International Journal of Computer Science and Information Technologies*, Vol.5 No.03 2014.
- Nguyen, T., Lucas, D., Krohn-Grimberghe, A., & Schmidt-Thieme, L. (2010). Recommender System for Predicting Student Performance. *Procedia Computer Science* 1 (2010)
- Nguyen, T., Busche, A., & Schmidt-Thieme, L. (2009), Improving Academic Performance Prediction by Dealing with Class Imbalance. 2009 Ninth International Conference on Intelligent Systems Design and Applications.
- Nikam, S. (2015). A Comparative Study of Classification Techniques in Data Mining Algorithms. *Oriental Journal of Computer Science and Technology* ;8(1), 2015, ISSN : 0974-6471 Online ISSN : 2320-8481
- Nikolovski, V., Stojanov, R., Mishkovski, I., Chorbev, I., & Madjarov., G. (2015). Educational Data Mining: Case Study for Predicting Student Dropout in Higher Education. <https://www.researchgate.net/publication/282333827> Conference Paper · April 2015

- Nisbet R., Elder, J., & Miner, G. (2009). Handbook of statistical analysis and data mining applications. 1st ed. Amsterdam: Academic Press/Elsevier, 2009.
- Nithya, P., Umamaheswari, B., & Umadevi, A. (2016). A survey on educational data mining in field of education. *J Comput Sci Softw Dev* 1: 1-6.
- Nitya, Upadhyay., Vinodini, K. (2014). A Survey on the International Journal of Computer Applications Technology and Research. Vol.3 No.11, 2014
- Nizar, A., Dong, Z., Wang, Y., (2008). Power utility nontechnical loss analysis with extreme learning machine method. *Power Systems, IEEE Transactions on*, 2008; 23: 946-955.<https://doi.org/10.1109/TPWRS.2008.926431>
- Noha, H., & Saif, E. (2015). Predict Academic Performance of Students using an K Nearest Neighbour Algorithm Case Study: MATLAB Course. *International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064 Index Copernicus Value (2015): 78.96 | Impact Factor (2015): 6.391*
- Ogor E. (2007). Student Academic Performance Monitoring and Evaluation Using Data Mining Techniques. Conference: Electronics, Robotics and Automotive Mechanics Conference, 2007. CERMA 2007: DOI: 10.1109/CERMA.2007.4367712
- Ogwoka, T., Cheruiyot, W., & Okeyo, G. (2015). A Model for Predicting Students' Academic Performance using a Hybrid of K-means and Decision tree Algorithms. *International Journal of Computer Applications Technology and Research* Volume 4– Issue 9, 693 - 697, ISSN: 2319–8656, 2015.
- Omisore, O., & Azeez, N. (2016). Predicting Academic Performance of Students with KNN Classifier. Conference: ACM International Conference on Computer Science Research and Innovations (CoSRI 2015)
- Osmanbegovic, E., & Suljic, M., (2012). Data mining approach for predicting student performance. *Economic Review Journal of Economics and Business*. Volume 10(1), 2012.
- Osuna, R., (2002). Lecture Notes CS 790: Introduction to Pattern Recognition. Wright State University, Dayton, Ohio, USA.
- Pandey, M., & Sharma, V., (2013), A Decision Tree Algorithm Pertaining To The Student Performance Analysis and Prediction, *International Journal Of Computer Applications* (0975 – 8887) Volume 61– No.13, January 2013.
- Pandey, M., & Taruna, S. (2014). An Empirical Analysis of Classification Techniques for Predicting Academic Performance. 10.1109/IAdCC.2014.6779379.

- Balaji, P & Srinivasan, D. (2010). An Introduction to Multi-Agent Systems. 10.1007/978-3-642-14435-6\_1.
- Paris. I., Affendey, L., & Mustapha, N.(2010). Improving academic performance prediction using voting technique in data mining. *World Academy of Science, Engineering and Technology*, vol 4, pp. 820--823, 2010.
- Patil, D., Wadhai, V., & Gokhale, J., (2010). Evaluation of decision tree pruning algorithms for complexity and classification accuracy 2010.Quinlan JR. *Induction of decision trees. Machine learning* 1986; 1: 81-106. <https://doi.org/10.1007/BF00116251>
- Patil, V., Suryawanshi, S., Saner, M., Patil, V., & Sarode, B. (2017), Student performance prediction using classification data mining techniques. *International Journal of Scientific Development and Research* 2(6):163–167, 2017.
- Pascarella, E., & Terenzini, P. (2005). *How college affects students. A third decade of research.* San Francisco: Jossey-Bass.
- Phyu, T., (2009). Survey of Classification Techniques in Data Mining. *Int. Multi-conference Eng. Computer. Sci.*, vol.I, pp. 18–20, 2009.
- Pinkas, B., (2002). Cryptographic techniques for privacy-preserving data mining. *ACM SIGKDD Explorations Newsletter*, 4(2):12–19, 2002.
- Rafaeilzadeh, P., Tang, L., & Liu, H. (2009), *Cross Validation*, *Encyclopedia of Database Systems*, 2009.
- Quadri, M., & Kalyankar, N. (2010). Drop Out Feature of Student Data for Academic Performance Using Decision Tree Techniques. *Global Journal of Computer Science and Technology*, Page | 2 Vol. 10 Issue 2 (Ver 1.0), April 2010
- Quinlan, J., (1986). Induction of decision trees. *Machine learning*. 1986; 1: 81-106. <https://doi.org/10.1007/BF00116251+A17>
- Quinlan, J. (1987). Simplifying decision trees. *International Journal of man-Machine Studies* 1987; 27: 221-234.[https://doi.org/10.1016/S0020-7373\(87\)80053-6](https://doi.org/10.1016/S0020-7373(87)80053-6)
- Ragab, A., Mashat, A., Khedra, A. (2012). HRSPCA: Hybrid Recommender System for Predicting College Admission. *Intelligent Systems Design and Applications (ISDA)*, 2012 12<sup>th</sup> International Conference, PP107-113, Kochi, India, 27-29 Nov.2012.
- Ramaswami, M., & Bhaskaran, R. (2010). A CHAID Based Performance Prediction Model in Educational Data Mining. *IJCSI International Journal of Computer Science Issues*, Vol. 7, Issue 1, No. 1, January 2010 ISSN (Online): 1694-0784 ISSN (Print): 1694-0814

- Ramesh, V., Parkavi, P., & Ramar, K. (2013). Predicting Student Performance: A Statistical and Data Mining Approach. *International Journal of Computer Applications*. 63.975-8887
- Rangra, K., & Bansal, D. (2014). "Comparative Study of Data Mining Tools.. *International Journal of Advanced Research in Computer Science and Software Engineering*. Volume 4, Issue 6, June 2014 ISSN: 2277 128X. www.ijarcsse.com
- Rathee, A., & Mathur, R. (2013). Survey on Decision Tree Classification algorithms for the Evaluation of Student Performance, *International Journal of Computers & Technology*, vol 4, iss 2, pp. 244--247, 2013.
- Rauf, A., Mahfooz, S., Khusro, S., & Javed, H., (2012). Enhanced K-Mean Clustering Algorithm to Reduce Number of Iterations and Time Complexity. *Middle-East Journal of Scientific Research*. 12. 959-963. 10.5829/idosi.mejsr.2012.12.7.1845.
- Romero, C., & Ventura, S. (2010). Educational Data Mining: A Review of the State of the Art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40(6), pp.601–618.
- Romero, C., & Ventura, S., (2007). Educational data mining: A survey from 1995 to 2005. *Expert Systems with Applications*. 33. 135-146. 10.1016/j.eswa.2006.04.005.
- Romero, C., & Ventura, S. (2013). Data Mining in Education. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*. 3. 10.1002/widm.1075.
- Rutkowski, L., Pietruczuk, L., Duda, P., Jaworski, M. (2013). Decision trees for mining data streams based on the McDiarmid's bound. *Knowledge and Data Engineering, IEEE Transactions on*, 2013; 25: 1272-1279. <https://doi.org/10.1109/TKDE.2012.66>
- Saputra, M., Widiyaningtyas, T., Prasetya, W. (2018). Illiteracy Classification Using K Means-Naïve Bayes Algorithm. *International Journal of Informatics Visualization*. Vol 2 (2018) No 3 e-ISSN : 2549-9904 ISSN : 2549-9610
- Sagardeep, R., & Garg, A. (2017). Analyzing Performance of Students by Using Data Mining Techniques A Literature Survey. 2017 4th IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics (UPCON)
- Sajadin, S., Zarlis, M., Hartama, D., Ramliana, S., & Wani, E. (2011). Prediction Of Student Academic Performance by An Application of Data Mining Techniques. 2011 International Conference On Management And Artificial Intelligence Ipedr Vol.6 (2011) pp. 110 -114.

- Sarker, F., Thanassis, T., Hugh, C. (2013). Student's performance prediction by using institutional internal and external open data sources. CSEDU: 5th International Conference on Computer Supported Education, Germany.
- Sen, B., & Ucar, E. (2012). Evaluating the achievements of computer engineering department of distance education students with data mining methods. *Procedia Technology* 1 262 – 267, 2012.
- Serrano, E., Rovatsos, M., & Botía, J. (2015). Data mining agent conversations: A qualitative approach to multiagent systems analysis. *Journal of Theoretical and Applied Information Technology*, vol. 73 No. 3, pp. 132-146, March 2015.
- Shahiria, A., Wahidah, H., & Nur'aini, A. (2015). The Third Information Systems International Conference A Review on Predicting Student's Performance using Data Mining Techniques . *Procedia Computer Science* Volume 72, 2015, Pages 414-422
- Sharma, S., Agrawal, J., Agarwal, S. (2013). Machine learning techniques for data mining: A survey, in *Computational Intelligence and Computing Research (ICCIC)*. 2013 IEEE International Conference on, 2013; pp. 1-6.
- Shovon, H. (2012). Prediction of Student Academic Performance by an Application of K-Means Clustering Algorithm, *International Journal of Advanced Research in Computer Science and Software Engineering*, Vol. 2(7), July 2012.
- Siemens, G., & Baker, R. (2012). Learning analytics and educational data mining: towards communication and collaboration. In *Proceedings of the 2nd international conference on learning analytics and knowledge*, pages 252-254. ACM.
- Sin, K., & Muthu L. (2015), Application of big data in education data mining and learning analytics-A literature review, *Ictact J. Soft Computing Spec. Issue Soft Computing Model. Big Data*, vol. 5, no. 4, pp. 1035–1049, 2015
- Srinivasan, J. (2015). A Study on Role of Intelligent Agent in Education. *International Journal of Applied Engineering Research*. 10. 497-501.
- Sivasakthi, M. (2017). Classification and Prediction based Data Mining Algorithms to Predict Students' Introductory programming Performance. *Proceedings of the International Conference on Inventive Computing and Informatics (ICICI 2017)*
- Sukanya, M, Biruntha, S., Karthik, S., & Kalaikumaran, T., (2012). Data Mining: Performance Improvement in Education Sector using Classification and Clustering. *International Conference on Computing and Control Engineering (ICCCE)*, 2012.

- Sumitha, R., & Vinothkumar, E. (2016). Prediction of Student Outcomes using Data Mining Techniques. *International Journal of Scientific Engineering and Applied Science*, Vol.2, Issue, pp. 132-139, 2016.
- Sun. H., (2010). Research on Student Learning Result System based on Data Mining. *International Journal of Computer Science & Network Security*. Vol. 10, no. 4, pp. 203–205, 2010.
- Swecker, H., Fifolt, M., & Searby, L. (2013). Academic advising and first-generation college students: A quantitative study on student retention. *NACADA Journal*, 33(1), 46-53
- Tekin, A. (2014). Early Prediction of Students' Grade Point Averages at Graduation: A Data Mining Approach. *Eurasian Journal of Educational Research*, Issue 54, 2014, pp. 207-226
- Timofte, R., Tuytelaars, T., & Van G. (2012). Naive bayes image classification: beyond nearest neighbors. Proceedings of 11th Asian conference on computer vision - ACCV 2012, November 5-9, 2012, Daejeon, Korea.
- Tsai, C., & Chen M. ( 2010). Credit Rating by Hybrid Machine Learning Techniques. *Applied Software Computing*. Vol. 10, pp. 374–380, 2010.
- Twa, M., Parthasarathy, S., Roberts, C., Mahmoud, A., Raasch. T., & Bullimore, M. (2005). Automated decision tree classification of corneal shape. *Optometry and vision science: official publication of the American Academy of Optometry* 2005; 82: 1038.<https://doi.org/10.1097/01.opx.0000192350.01045.6f>
- Ulyani, N., Mohd, N., Nor-Aini, Y., & Amin A. (2017). Service Quality Performance of Student Housing: The Effects on Students Behavioural Intentions. 2017 IEEE 15th Student Conference on Research and Development (SCORED)
- Undavia, J., Patel, A., Dolia, P. (2013). Comparison of Classification Algorithms to Predict Students' Post Graduation Course in Weka Environment. *International Journal of Advanced Research in Computer Science and Software Engineering*, Volume 3, Issue 9, September 2013.
- Upadhyay, N., & Katiyar, V., (2014). A Survey on the Classification Techniques in Educational Data Mining. *International Journal of Database Management System (IJDMS)*, 3(11), 2014, 725–728.
- Vaidya, J., (2003). Privacy Preserving K-Mean Clustering over Vertically Partitioned Data. Proceeding of SIGkDD'03, Washington, DC, USA, August 24-27, 2003
- Vandamme, J., Meskens, N., & Superby, J. (2007). Predicting Academic Performance by Data Mining Methods. *Education Economics*, Volume 15, No. 4, 2007.

- Ward, A., Howard, S., & Murray-Ward, M. (1996). Achievement and Ability Tests - Definition of the Domain. *Educational Measurement*, 2, University Press of America, pp. 2–5, ISBN 978-0-7618-0385-0
- Walid M., Osama, F., & Heba, M. (2013). Automated Student Advisory using Machine Learning. *International Journal of Computer Application (IJCA)*, Nov 2013.
- Wallace, C., Korb, K., & Dai, H. (1996). Causal discovery via mml. *International Conference for Machine Learning (ICML)*, vol. 96, pp. 516–524, Citeseer, 1996.
- Wang, C., (2006). New ensemble machine learning method for classification and prediction on gene expression data. pp.3478--3481, 2006.
- Winston, R., & Sandor, J. (1984). Developmental academic advising: What do students want??. *National Academic Advising Association [NACADA] Journal*, 4(1), 5-13.
- Witten, I., & Frank, E. (2005). *Data Mining: Practical machine Learning tools and techniques*. 2nd edition, Morgan Kaufmann, San Francisco(2005).
- Woodman, R., (2001). Investigation of factors that influence student retention and success rate on Open University courses in the East Anglia region. M.Sc. Dissertation, Sheffield Hallam University, UK.
- Wooldridge, M., & Jennings, N. (1995). Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, 10(2), 115-152. doi:10.1017/S0269888900008122
- Wu, X., Kumar, V., Quinlan, J., Ghosh, J., Yang, Q., & Motoda, H. (2008). Top 10 algorithms in data mining. *Knowledge and Information Systems* 2008; 14: 1-37. <https://doi.org/10.1007/s10115-007-0114-2>
- Xiaofeng, M., & Zhurong, Z. (2018). Student Pass Rates Prediction Using Optimized Support Vector Machine and Decision Tree, 978-1-5386-4649-6/18/\$31.00. ©2018 IEEE.
- Xie, Z., Hsu, W., Liu, Z., & Mong-Li, L. (2002). SNNB: A Selective Neighborhood Based Naïve Bayes for Lazy Learning. *PAKDD*, 104-114.
- Yadav, S., Kumar, B., & Saurabh P., (2012). Mining education data to predict student's retention: A comparative study. *International Journal of Computer Science and Information Security* 10(2), 113-117.
- Yadav, S., Bharadwaj, B., & Pal S. (2012). Mining Education data to predict student's retention: a comparative study. arXiv preprint arXiv:1203.2987, 2012.
- Yahia, E., Eldow, M., & El-Mukashfi, E. (2010). A New Approach for Evaluation of Data Mining Techniques. *International Journal of Computer Science Issues*. 7. 181-186.



- Yang, Y., & Webb, G. (2009). Discretization for naive-Bayes learning: managing discretization bias and variance. *Machine learning* 2009; 74: 39-74. <https://doi.org/10.1007/s10994-008-5083-5>
- Yanga, F., & Frederick, W. (2018). Study on student performance estimation, student progress analysis, and student potential prediction based on data mining. *Computers & Education* 123 (2018) 97–108
- Yassein, N., Helali, R., & Mohomad, S., (2017). Predicting Student Academic Performance in KSA using Data Mining Techniques. *Journal for Information Technology and Software Engineering* 7: 213. DOI: 10.4172/2165-7866.1000213
- Yofi. F., Riza. A., & Much. A. (2018). K-Nearest Neighbor and Naive Bayes Classifier Algorithm in Determining The Classification of Healthy Card Indonesia Giving to The Poor. *Scientific Journal of Informatics* Vol. 5, No. 1, May 2018. p-ISSN 2407-7658 <http://journal.unnes.ac.id/nju/index.php/sji> e-ISSN 2460-0040
- Young-Jones, A., Burt, T., Dixon, S., & Hawthorne, M. (2013). Academic advising: does it really impact student success?. *Quality Assurance in Education*, 21(1), 7-19.
- Yu, C., DiGangi, A., Jannasch, P., & Kaprolet, C. (2010). A Data Mining Approach for Identifying Predictors of Student Retention from Sophomore to Junior Year". – *Journal of Data Science*, Vol. 8, 2010, 307-325.
- Ziedner, M. (1998). *Test anxiety: The state of the art*". New York: New York: Plenum Press. p. 259. ISBN 9780306471452. OCLC 757106093

## Source Code

```
/*  
 * To change this template, choose Tools | Templates  
 * and open the template in the editor.  
 */  
  
packageMultiAgentPlatform;  
  
importDBConnection.DatabaseConnection;  
  
importjava.awt.BorderLayout;  
  
importjava.awt.Color;  
  
importjava.io.BufferedReader;  
  
importjava.io.File;  
  
importjava.io.FileInputStream;  
  
importjava.io.FileNotFoundException;  
  
importjava.io.FileOutputStream;  
  
importjava.io.FileReader;  
  
importjava.io.IOException;  
  
importjava.io.ObjectInputStream;  
  
importjava.io.ObjectOutputStream;  
  
importjava.sql.Connection;  
  
importjava.sql.DriverManager;  
  
importjava.sql.PreparedStatement;  
  
importjava.sql.ResultSet;  
  
importjava.sql.SQLException;  
  
importjava.util.ArrayList;  
  
importjava.util.Calendar;
```

```
import java.util.GregorianCalendar;

import java.util.List;

import java.util.Random;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.DefaultListModel;

import javax.swing.JFileChooser;

import javax.swing.JFrame;

import javax.swing.JOptionPane;

import javax.swing.JTextField;

import javax.swing.SpinnerNumberModel;

import javax.swing.table.DefaultTableModel;

import org.jfree.ui.RefineryUtilities;

import weka.classifiers.Classifier;

import weka.core.converters.ConverterUtils.DataSource;

import weka.classifiers.Evaluation;

import weka.classifiers.evaluation.NominalPrediction;

import weka.core.FastVector;

import weka.core.Instances;

import weka.classifiers.trees.J48;

import weka.classifiers.trees.REPTree;

import weka.classifiers.lazy.IBk;

import weka.classifiers.bayes.NaiveBayes;

import weka.classifiers.bayes.NaiveBayesMultinomial;

import weka.classifiers.evaluation.ThresholdCurve;
```

```

importweka.classifiers.functions.SMO;

importweka.classifiers.meta.FilteredClassifier;

importweka.core.Attribute;

importweka.core.DenseInstance;

importweka.core.Utils;

importweka.core.converters.ArffLoader;

importweka.core.converters.ArffSaver;

importweka.core.tokenizers.NGramTokenizer;

importweka.filters.unsupervised.attribute.Remove;

importweka.filters.unsupervised.attribute.StringToWordVector;

import weka.gui.visualize.PlotData2D;

importweka.gui.visualize.ThresholdVisualizePanel;

/**
 *
 * @author ORAH RICHARD
 */

public class MultiAgentStudentPrediction extends javax.swing.JFrame {

    Connection conn;

    PreparedStatementpst;

    ResultSetsrs;

    private static final long serialVersionUID = 1L;

    File file1;

    JFileChooserjfc = new JFileChooser();

    DefaultListModelwordModel;

```

```

DefaultListModel predModel;
DefaultListModel goldModel;
int lock = 0;

    //NOTE FOR FINAL

private static final Logger LOGGER = Logger.getLogger("MultiAgentStudentPrediction ");

private FilteredClassifier classifier;

    //declare train and test data Instances

private Instances trainData;

private ArrayList< Attribute> wekaAttributes;

    Connection connection = null;

    /**
     * Creates new form MultiAgentStudentPrediction
     */

public MultiAgentStudentPrediction() {
    getContentPane().setBackground(Color.pink);
    initComponents();
    connection = DatabaseConnection.dbConnector();

    // String curDir = System.getProperty("user.dir");

    currentDate();

    cl();

```

```

setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);

getContentPane().setLayout(null);

setTitle("STUDENT PERFORMMANCE PREDICTION USING MULTI-AGENT DATA MINING");

setResizable(false);

setVisible(true);

        // Progresslabel.setText("Please wait...");

SpinnerNumberModelnum_model = new SpinnerNumberModel(100, 1, 10000000, 1);

spMaxItrs.setModel(num_model);

        String curDir = System.getProperty("user.dir");

        //txtInputCoNLLFile.setText(curDir + File.separator + "sampledata" + File.separator +
"class_label_Illegal" + File.separator + "");

        //txtLegalMessage.setText(curDir + File.separator + "sampledata" + File.separator +
"class_label_legal" + File.separator + "");

        //txtSampleTextFileConcatenate.setText(curDir + File.separator + "sampledata" + File.separator +
"Concatenate_Both_Two_Class" + File.separator + "DataAnalytics.arff");

        //TxtSampleFileExcell.setText(curDir + File.separator + "sampledata" + File.separator +
"Excell_Sample" + File.separator + "Trainingbigdata.arff");

txtTrainFile.setText(curDir + File.separator + "Data" + File.separator + "Cross-validation-set2.arff");

txtTraining.setText(curDir + File.separator + "Data" + File.separator + "Training.arff");

txtTesting.setText(curDir + File.separator + "Data" + File.separator + "Testing.arff");

TRAIN_DATA.setText(curDir + File.separator + "dataset" + File.separator + "train.txt");

```

```

TRAIN_ARFF_ARFF.setText(curDir + File.separator + "dataset" + File.separator + "train.arff");
TEST_DATA.setText(curDir + File.separator + "dataset" + File.separator + "test.txt");
TEST_DATA_ARFF.setText(curDir + File.separator + "dataset" + File.separator + "test.arff");
jTextField1.setText(curDir + File.separator + "models" + File.separator + "model.dat");

    //txtTrainFile.setText(curDir + File.separator + "data" + File.separator + "ModelTrainingSet.arff");

    // txtLegalMessage.setText(curDir + File.separator);

    ///txtOutputDepPath.setText(curDir + File.separator + "sampledata" + File.separator +
"Excell_Sample" + File.separator + "SampleDataBinaryFile.xls");

//Newly Past File

//txtModelPath1.setText(curDir + File.separator + "dataset" + File.separator + "test.arff");

// txtInputFile1.setText(curDir + File.separator + "dataset" + File.separator + "test.txt");

//txtInputFile2.setText(curDir + File.separator + "dataset" + File.separator + "train.arff");

//txtInputFile3.setText(curDir + File.separator + "dataset" + File.separator + "train.txt");

// txtInputTagFile.setText(curDir + File.separator + "data" + File.separator + "tag" + File.separator +
"gold_test.lbl");

// txtInputCoNLLFileT2D.setText(curDir + File.separator + "data" + File.separator + "conll" +
File.separator + "test.conll");

//txtTrainFile.setText(curDir + File.separator + "data" + File.separator + "tag" + File.separator +
"train.lbl");

// txtTrainModelPath.setText(curDir + File.separator + "model" + File.separator);

//txtModelPath.setText(curDir + File.separator + "data" + File.separator + "training_set" +
File.separator + "TrainingSet.arff");

```

```
//txtInputFile.setText(curDir + File.separator + "data" + File.separator + "test_set" + File.separator + "TestingSet.arff");
```

```
//txtGoldFile.setText(curDir + File.separator + "data" + File.separator + "tag" + File.separator + "gold_test.lbl"
```

```
/*
```

```
* Class for running an arbitrary classifier on data that has been passed through an arbitrary filter
```

```
* Training data and test instances will be processed by the filter without changing their structure
```

```
*/
```

```
classifier = new FilteredClassifier();
```

```
// set Hybridization of K-nn and NaiveBayes as arbitrary classifier
```

```
classifier.setClassifier(new NaiveBayes());
```

```
classifier.setClassifier(new IBk());
```

```
// Declare text attribute to hold the message
```

```
Attribute attributeText = new Attribute("text", (List< String>) null);
```

```
// Declare the label attribute along with its values
```

```
ArrayList< String>classAttributeValues = new ArrayList<>();
```

```
classAttributeValues.add("HIGH");
```

```
classAttributeValues.add("LOW");
```

```
classAttributeValues.add("MEDIUM");
```

```
Attribute classAttribute = new Attribute("CLASS", classAttributeValues);
```







login PanelModule



Figure : main menu Module

STUDENT PERFORMANCE PREDICTION USING MULTI-AGENT DATA MINING

File Student Account Registration Student/Admin Account Registration Course Form Registration

### STUDENT PERFORMANCE PREDICTION USING MULTI-AGENT DATA MINING

Home Page View Structured Dataset Model Training/Cross-Validation Training set/Test set Hybrid Model Performance Prediction/E-Adviser About

Dataset Structured DEMOGRAPHIC FACTORS ACADEMIC/WORK RELATED FACTOR SOCIAL FACTOR DATE:2019/6/23 TIME:3:0:18

Reg_No	Mode of	Gender	Marital	Program	Clb	Age	Emplo..	Family	Job Co.	Sponsor	Progra..	MAAC	Abst_L	Sub_A	Sub_B	Incomp.	Sub_H	Sub_C	Sub_U	Sub_Ek	
201448..	1	1	1	2	2	1	1	3	3	1	3	3	5	1	5	1	4	3	1	4	4
201448..	2	2	1	2	2	2	1	3	3	1	3	4	2	3	2	1	5	3	2	4	4
201448..	1	2	2	2	2	3	1	3	4	1	2	4	5	4	3	3	3	3	4	4	4
201448..	2	1	2	2	2	2	2	3	4	1	1	3	2	3	1	4	3	3	2	4	3
201448..	1	1	2	2	2	2	1	2	2	1	2	2	3	1	3	3	3	3	3	4	3
201548..	1	2	2	2	2	3	1	2	4	1	2	2	1	3	2	3	3	3	4	4	3
201548..	1	2	1	2	2	1	2	3	1	4	2	2	3	2	3	4	2	4	2	3	2
201548..	1	2	1	2	1	2	2	3	2	1	1	3	3	1	1	1	1	2	2	4	3
201548..	1	2	2	2	2	4	1	3	4	1	1	4	5	4	3	4	3	3	4	4	3
201548..	1	2	2	2	2	2	2	1	4	1	2	5	3	1	3	4	4	3	2	4	3
201548..	1	1	2	2	2	1	1	3	1	2	2	3	2	3	2	2	4	5	2	3	3
201548..	2	2	2	2	2	2	1	2	3	4	1	3	3	1	2	2	3	3	3	4	4
201548..	1	2	1	2	1	2	1	2	3	4	1	2	3	3	1	1	4	2	1	4	4
201548..	1	1	2	2	2	3	1	3	4	1	2	3	1	2	2	3	3	3	4	4	4
201548..	2	2	1	2	2	1	2	3	4	1	2	3	3	3	2	2	4	4	4	4	4
201548..	2	2	2	2	2	2	2	3	4	1	1	1	1	2	3	3	1	1	3	2	4
201548..	1	2	2	2	2	2	1	3	1	1	1	3	3	2	2	2	3	3	3	4	4
201548..	1	2	1	2	2	2	2	3	4	1	2	5	3	1	3	4	4	3	2	4	4
201548..	2	1	2	2	2	1	2	3	4	1	2	3	3	3	2	1	4	4	1	4	4
201548..	1	1	2	2	2	3	1	2	4	1	2	3	1	2	3	3	3	3	4	4	4
201548..	1	1	2	2	2	1	1	3	1	2	2	3	2	2	4	5	2	3	2	3	3
201548..	1	1	2	2	2	1	1	3	1	2	3	3	4	3	3	1	1	3	2	3	3
201548..	2	2	1	2	2	2	3	3	4	2	2	3	1	5	4	2	4	4	3	2	2
201548..	2	1	1	2	2	1	1	3	1	1	4	3	2	3	1	3	4	4	1	3	4

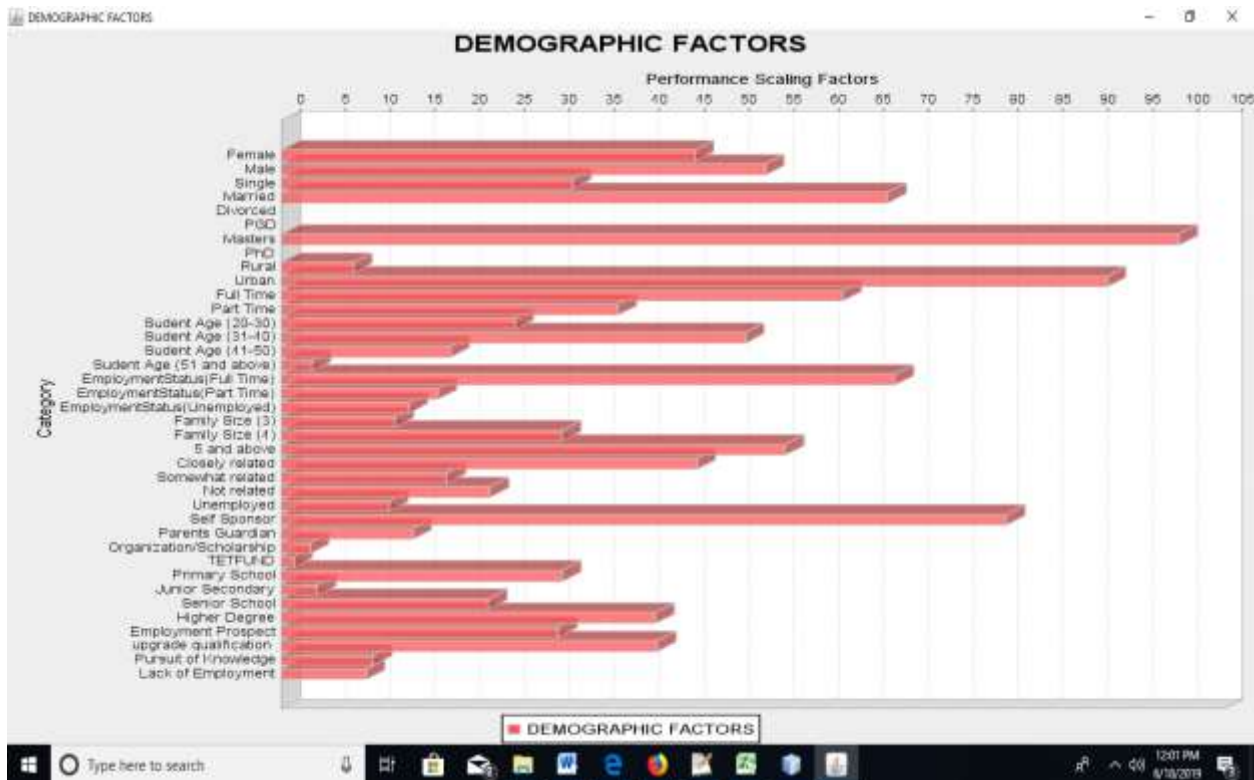
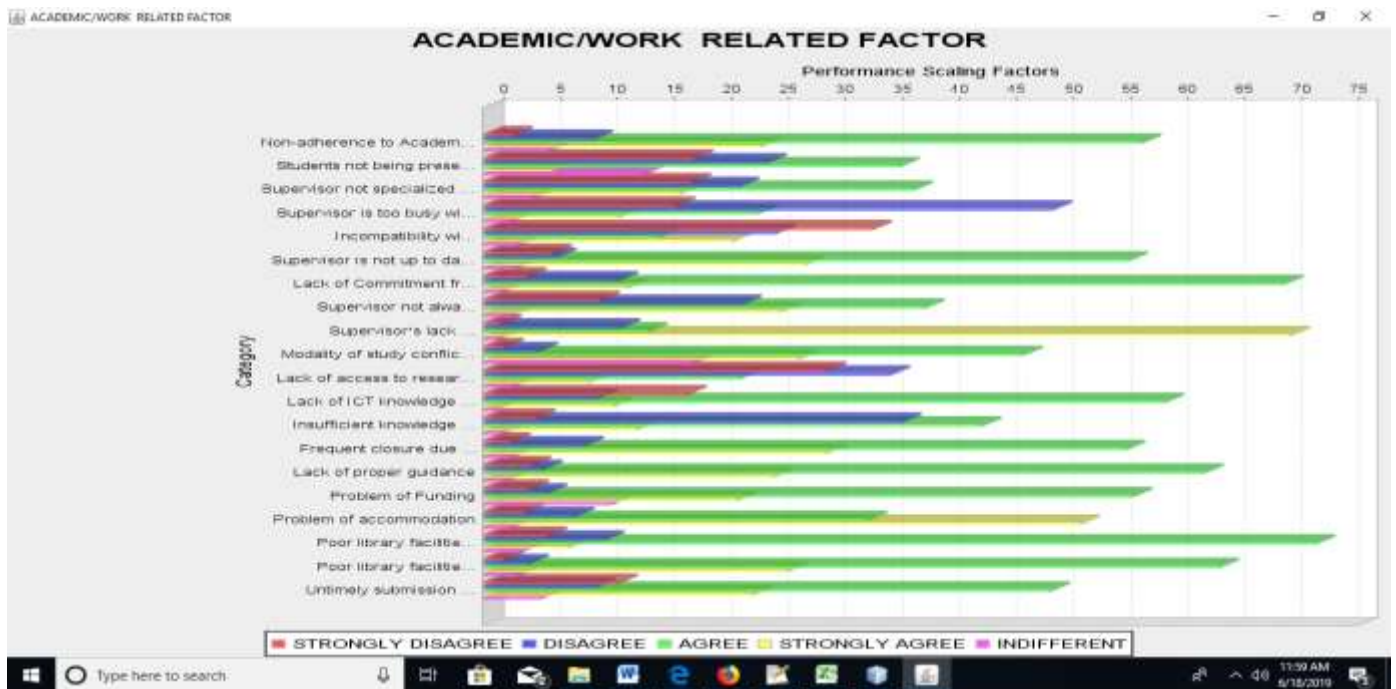
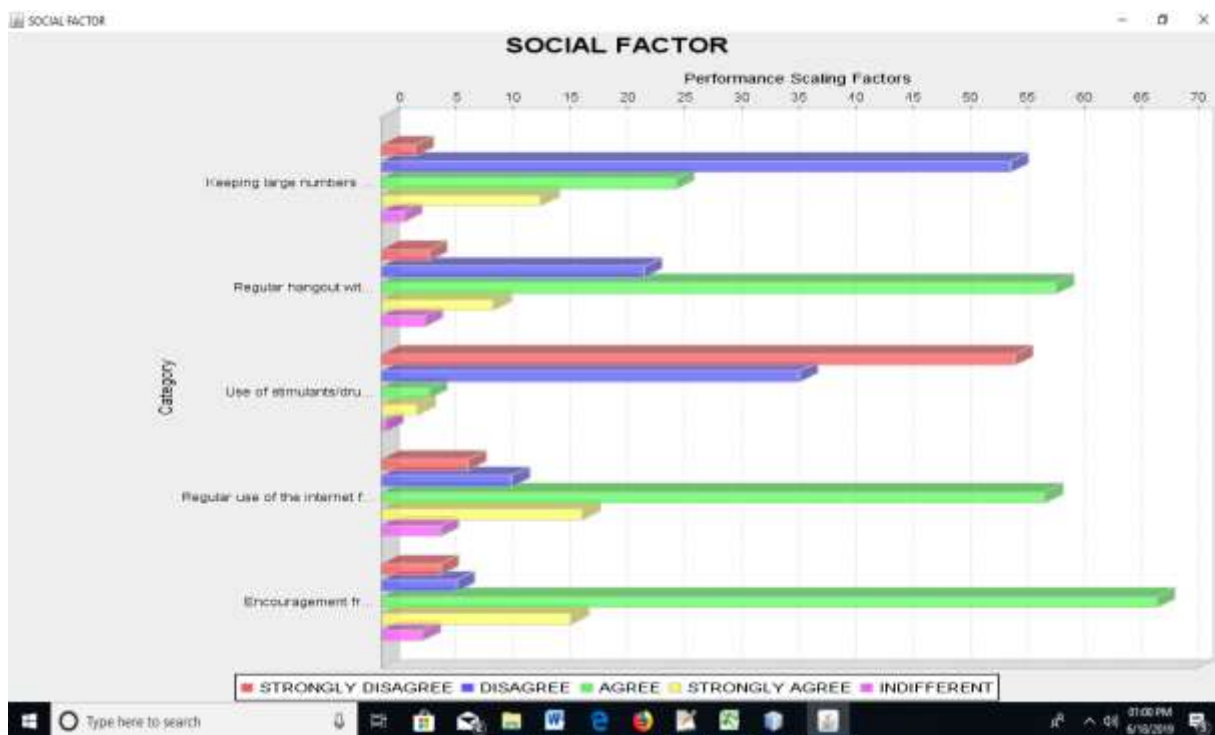


Figure: View Structured Dataset

Figure: Demographic Factors



**Academic/Work related factors**



**Social factors**

**USER ACCOUNT REGISTRATION**

UserID:  DATE: 2019/6/23

RegNo:  TIME: 4:19:27

Session:

Semester:

UserID	RegNo	Year	Role
1	2014406001F	2018/2019	FirstSemester
2	2014406002F	2018/2019	FirstSemester
3	2014406002F	2018/2019	FirstSemester
4	2015406035F	2019/2020	FirstSemester
5	2015406035F	2019/2020	SecondSemester
6	2017406085F	2019/2020	SecondSemester

**User Account Registration**

**STUDENT PERFORMANCE PREDICTION USING MULTI-AGENT DATA MINING**

**USER ACCOUNT REGISTRATION**

UserID:  DATE: 2019/7/15

User Name:  TIME: 11:29:33

Password:

Role Position:

UserID	Username	Password	Role
1	Richy	123	Admin
2	Fred	1234	Admin
3	Lucky	12345	PhD
4	Chigoda	12345	Msc
5	Dandadi	1234	Msc
6	Dandadi	2015406035F	Msc

**Admin Account Registration Module**

**Login Panel**

RegNo:

Session:

Semeter:

### Student Course Form Registration Menu

**REGISTER YOUR FIRST SEMESTER COURSE**

FULL NAME:  Current date: 23/02/2019 Time: 5:23:0

REGISTRATION NUMBER:

SESSION:

SEMESTER:

S/N	Select Course Code	Course Title	Credits Unit
1	<input type="text" value="Select Course"/>	Financial Accounting Theory	3
2	<input type="text" value="Select Course"/>	International Accounting	3
3	<input type="text" value="Select Course"/>	Auditing Theory	3
4	<input type="text" value="Select Course"/>	Management Accounting Theory	3
5	<input type="text" value="Select Course"/>	Public Sector Accounting and Finance	3
6	<input type="text" value="Select Course"/>	Advanced Reserch Methodology	3
7	<input type="text" value="Select Course"/>	Select one elective	3

Total Course:

## Student first semster Course Registration Form

**REGISTER YOUR SECOND SEMESTER MAIN COURSE/CARRY OVER COURSE**

SECOND SEMESTER COURSE REGISTRATION FORM    CARRY OVER FIRST SEMESTER COURSE FORM REGISTRATION    CARY OVER THIRD SEMESTER COURSE FORM REGISTRATION

FULL NAME:     Current date: 23/6/2019    Time: 8:25:53

REGISTRATION NUMBER:

SESSION:

SEMESTER:

S/N	Select Course Code	Course Title	Credits Unit
1	<input type="text" value="Select Course"/>	<input type="text" value="Taxation Theory and Practical"/>	<input type="text" value="2"/>
2	<input type="text" value="Select Course"/>	<input type="text" value="Corporate Finace"/>	<input type="text" value="2"/>
3	<input type="text" value="Select Course"/>	<input type="text" value="Management Information System"/>	<input type="text" value="2"/>
4	<input type="text" value="Select Course"/>	<input type="text" value="Economic Theory"/>	<input type="text" value="2"/>
5	<input type="text" value="Select Course"/>	<input type="text" value="Forensic Accounting"/>	<input type="text" value="2"/>
6	<input type="text" value="Select Course"/>	<input type="text" value="Environmental Accounting"/>	<input type="text" value="2"/>
7	<input type="text" value="Select Course"/>	<input type="text" value="Select one elective"/>	<input type="text" value="2"/>

   Total Course:

## Student Second semster Course Registration Form

**REGISTER YOUR THIRD SEMESTER MAIN COURSE/CARRY OVER COURSE**

THIRD SEMESTER COURSE REGISTRATION FORM    CARRY OVER FIRST SEMESTER COURSE REGISTRATION FORM    CARRY OVER THIRD SEMESTER COURSE REGISTRATION FORM

FULL NAME:     Current date: Label25    Time: Label25

REGISTRATION NUMBER:

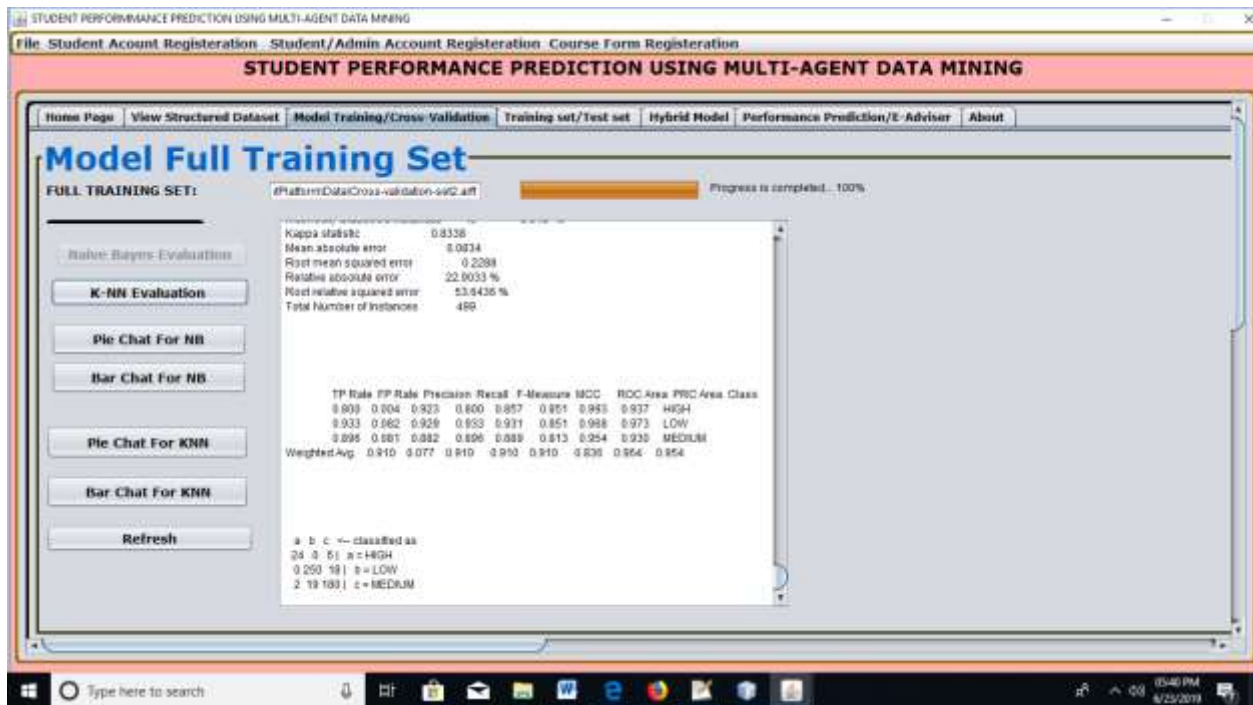
SESSION:

SEMESTER:

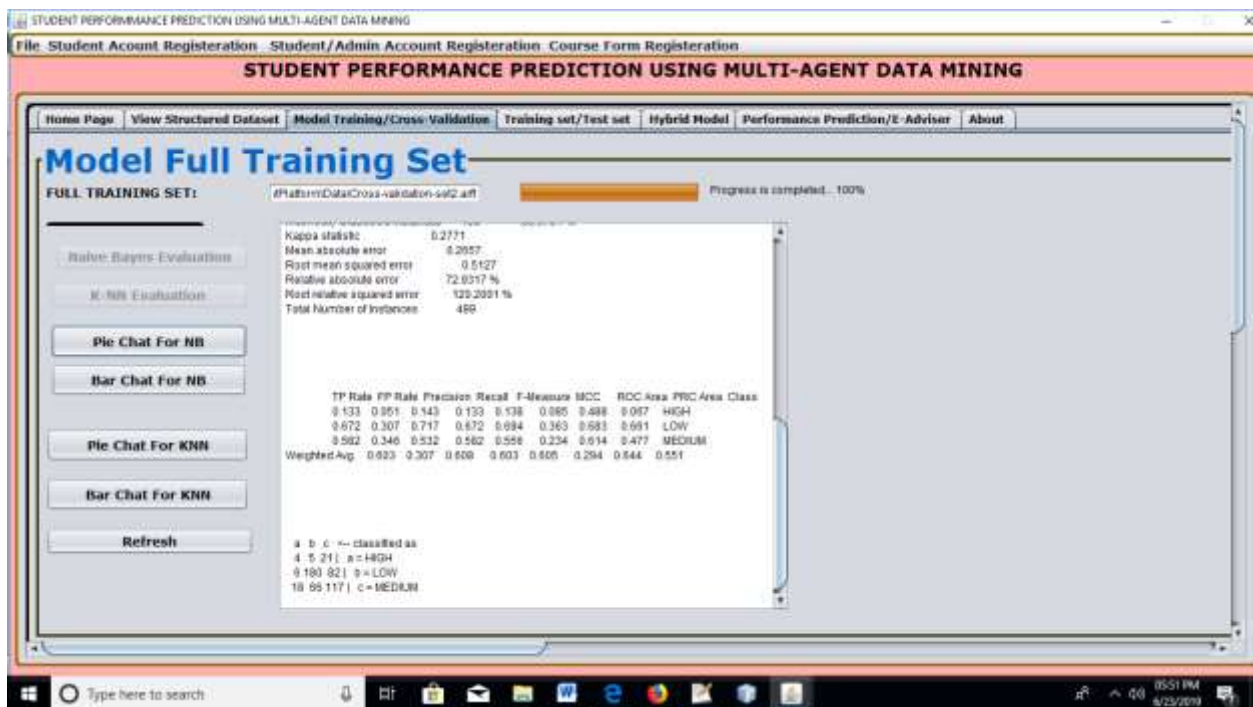
S/N	Select Course Code	Course Title	Credits Unit
1	<input type="text" value="Select Course"/>	<input type="text" value="Thesis"/>	<input type="text" value="12"/>
2	<input type="text" value="Select Course"/>	<input type="text" value="FIRST SEMESTER GRAND TOTAL"/>	<input type="text" value="21"/>
3	<input type="text" value="Select Course"/>	<input type="text" value="SECOND SEMESTER GRAND TOTAL"/>	<input type="text" value="21"/>

   Total Course:

## Student Third semster Course Registration Form

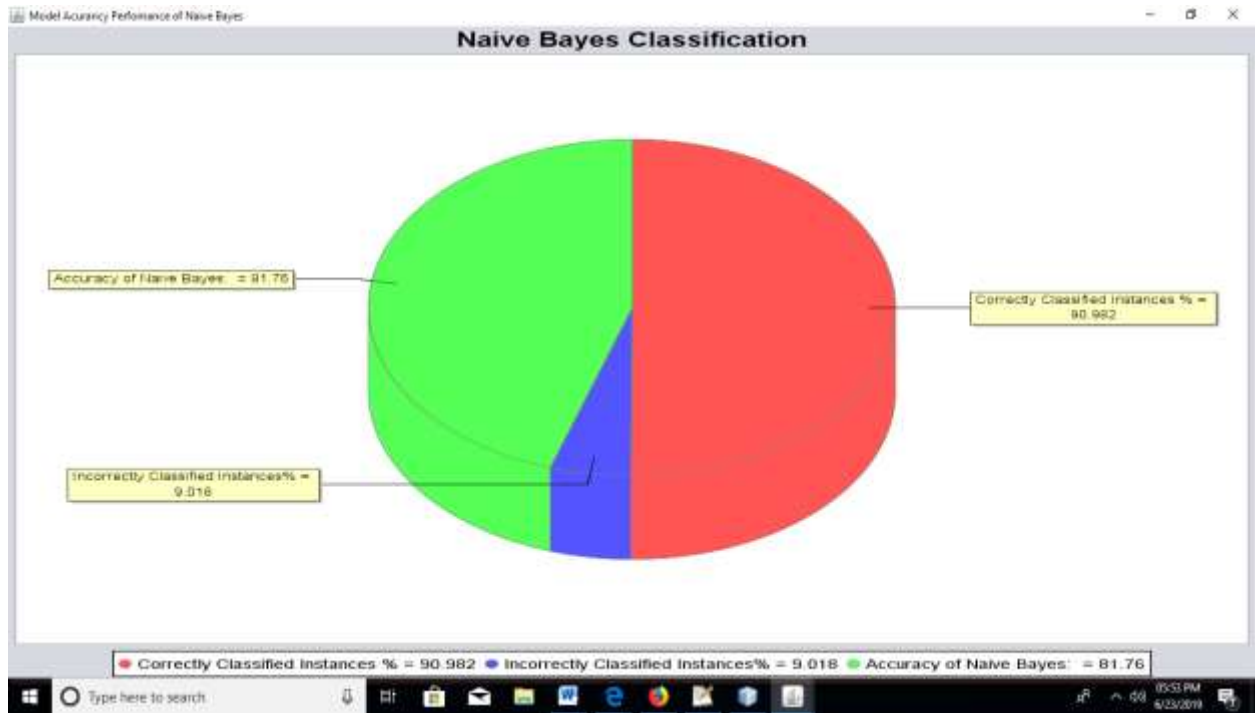


Model Building and Evaluation Interface for Naïve Bayes



Building and Evaluation Pie chat Interface for Naïve Bayes

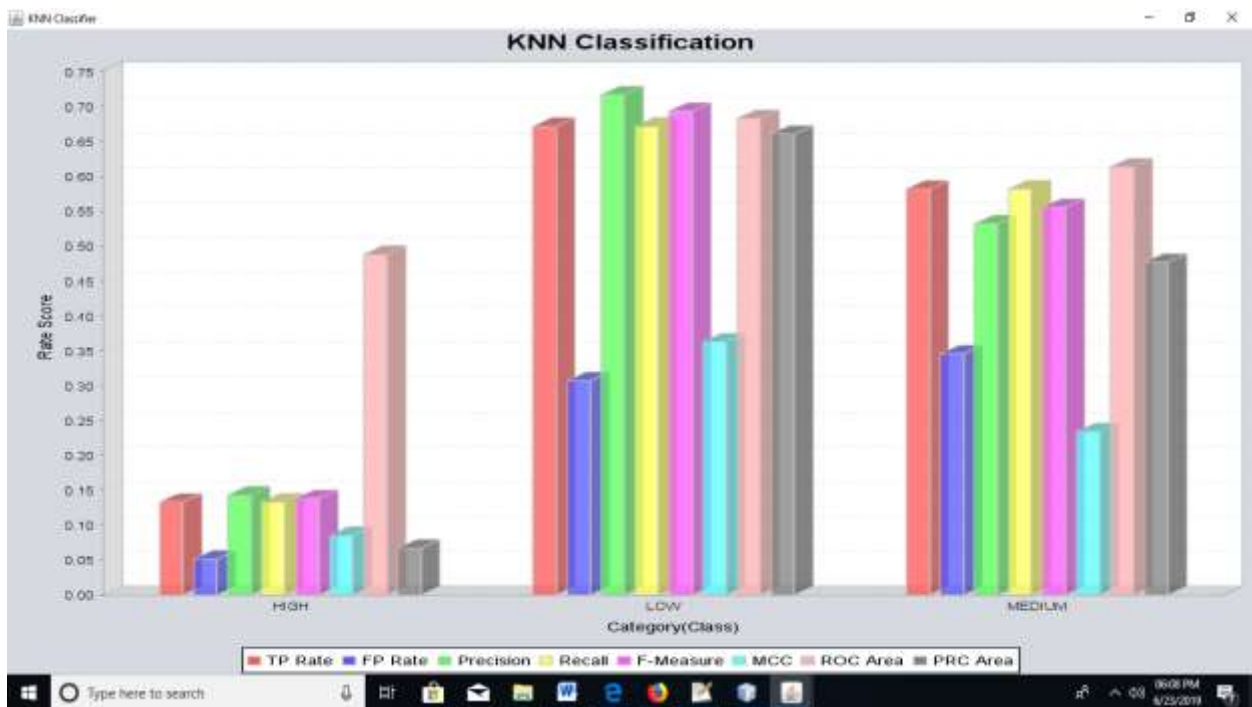
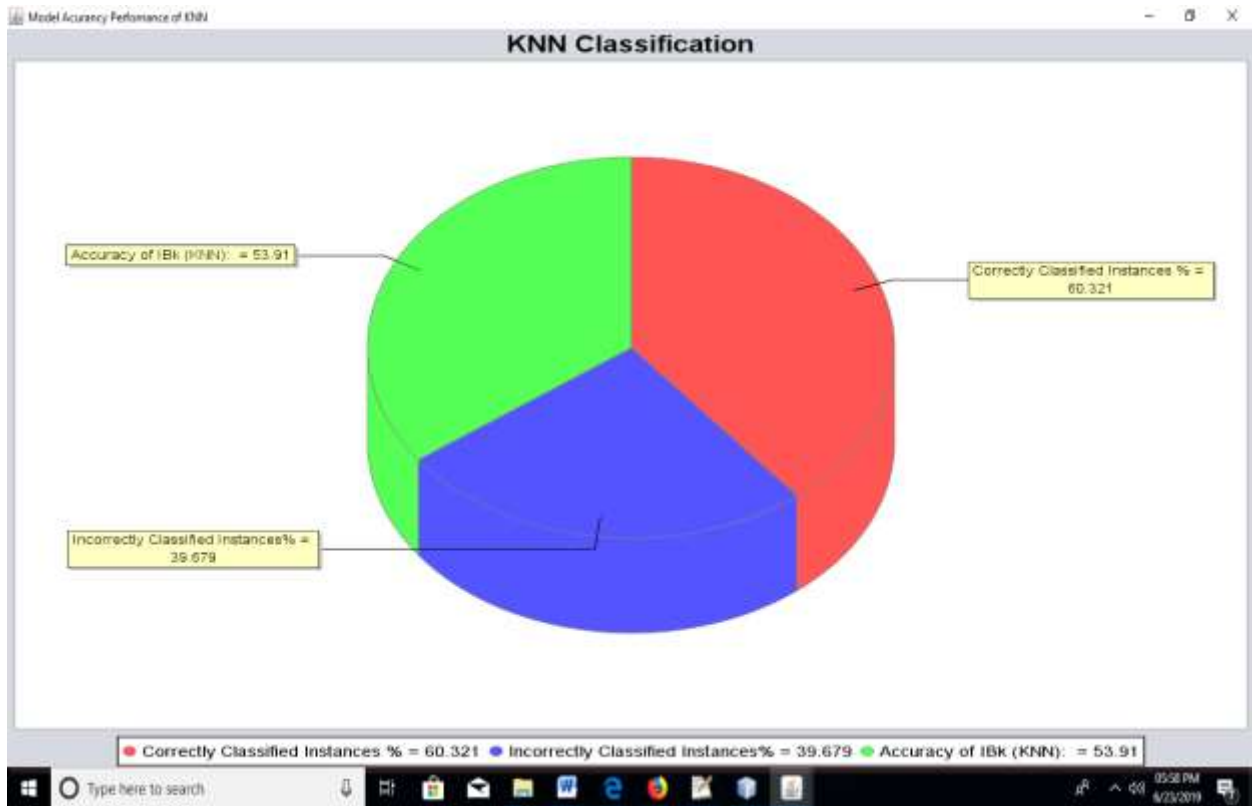




**Building and Evaluation Pie chat Interface for Naïve Bayes**



**Model Building and Evaluation bar chat Interface for Naïve Bayes**



STUDENT PERFORMANCE PREDICTION USING MULTI-AGENT DATA MINING

File Student Account Registration Student/Admin Account Registration Course Form Registration

### STUDENT PERFORMANCE PREDICTION USING MULTI-AGENT DATA MINING

Home Page View Structured Dataset Model Training/Cross-Validation Training set/Test set Hybrid Model Performance Prediction/E-Adviser About

**TRAINING SET** Projects\MultiAgentPlatform\Data\Training.sff **BROWSE TRAINING SET** ROC Curve Naive Bayes ROC Curve KNN

**TEST SET** IProjects\MultiAgentPlatform\Data\Testing.sff **BROWSE TESTING SET** Pie Chart Bar Chart Refresh

**Data Mining Classification** nb **Naive Bayes** **K-NN**

Complexity improvement (8) 350.5649 bits 0.7808 bits/instance  
 Mean absolute error 0.0710  
 Root mean squared error 0.2228  
 Relative absolute error 18.9972 %  
 Root relative squared error 52.6291 %  
 Total Number of Instances 449

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.917	0.012	0.815	0.917	0.863	0.895	0.886	0.820	HIGH
0.951	0.094	0.825	0.951	0.838	0.891	0.873	0.879	LOW
0.885	0.052	0.917	0.885	0.890	0.822	0.963	0.846	MEDIUM

Weighted Avg 0.815 0.073 0.918 0.815 0.915 0.845 0.971 0.963

a b c -- classified as  
 22 0 2 | a = HIGH  
 0 235 12 | b = LOW  
 5 19 154 | c = MEDIUM

Mean absolute error 0.1833  
 Root mean squared error 0.3995  
 Relative absolute error 80.2964 %  
 Root relative squared error 83.2855 %  
 Total Number of Instances 49

total\_instances : 49  
 correct pred. : 46  
 incorrect predictions : 3  
 accuracy : 71.43%

Mean absolute error 0.2188  
 Root mean squared error 0.4381  
 Relative absolute error 88.0045 %  
 Root relative squared error 103.8636 %  
 Total Number of Instances 49

total\_instances : 49  
 correct pred. : 49  
 incorrect predictions : 0  
 accuracy : 89.38%

Type here to search 12:10 PM 7/10/2009

STUDENT PERFORMANCE PREDICTION USING MULTI-AGENT DATA MINING

File Student Account Registration Student/Admin Account Registration Course Form Registration

### STUDENT PERFORMANCE PREDICTION USING MULTI-AGENT DATA MINING

Home Page View Structured Dataset Model Training/Cross-Validation Training set/Test set Hybrid Model Performance Prediction/E-Adviser About

**TRAINING SET** Projects\MultiAgentPlatform\Data\Training.sff **BROWSE TRAINING SET** ROC Curve Naive Bayes ROC Curve KNN

**TEST SET** IProjects\MultiAgentPlatform\Data\Testing.sff **BROWSE TESTING SET** Pie Chart Bar Chart Refresh

**Data Mining Classification** knn **Naive Bayes** **K-NN**

Complexity improvement (8) 350.5649 bits 0.7808 bits/instance  
 Mean absolute error 0.1230  
 Root mean squared error 0.348  
 Relative absolute error 34.4313 %  
 Root relative squared error 82.2107 %  
 Total Number of Instances 449

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.583	0.015	0.630	0.583	0.609	0.598	0.773	0.415	HIGH
0.874	0.198	0.864	0.874	0.869	0.707	0.847	0.813	LOW
0.770	0.148	0.774	0.770	0.772	0.623	0.807	0.717	MEDIUM

Weighted Avg 0.817 0.182 0.818 0.817 0.817 0.957 0.927 0.754

a b c -- classified as  
 14 1 9 | a = HIGH  
 0 216 31 | b = LOW  
 6 33 137 | c = MEDIUM

Mean absolute error 0.1833  
 Root mean squared error 0.3995  
 Relative absolute error 80.2964 %  
 Root relative squared error 83.2855 %  
 Total Number of Instances 49

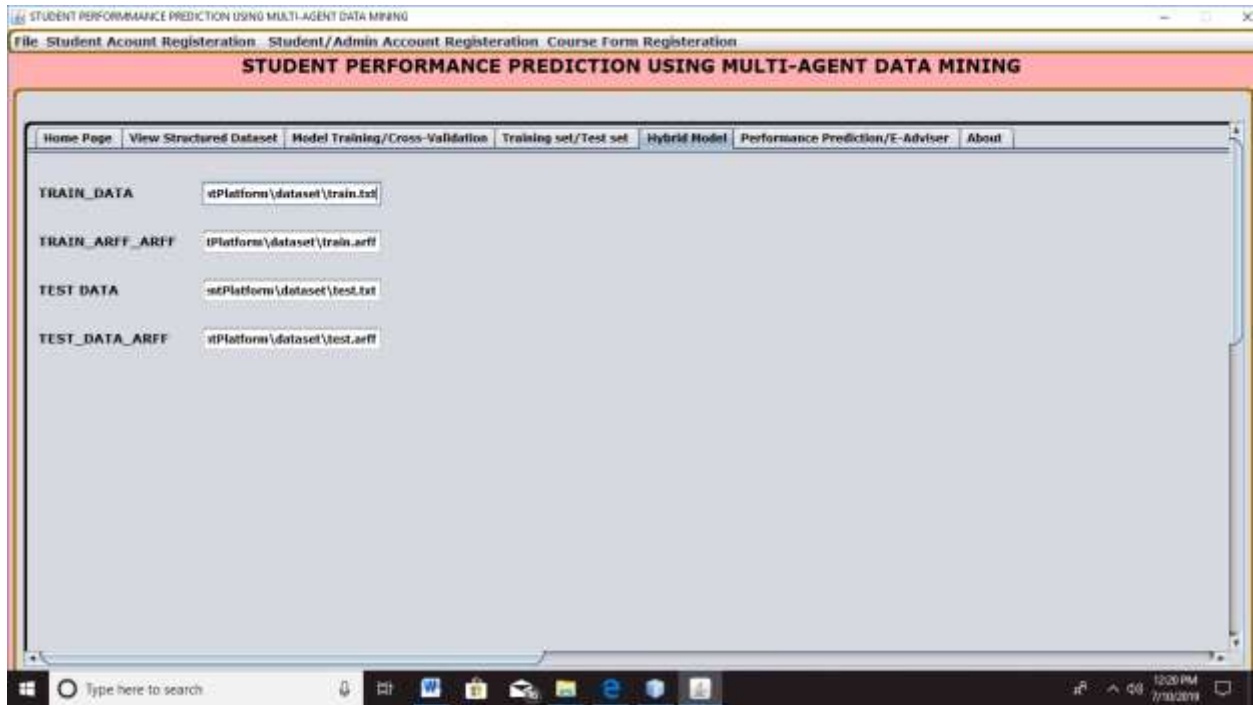
total\_instances : 49  
 correct pred. : 46  
 incorrect predictions : 3  
 accuracy : 71.43%

Mean absolute error 0.2188  
 Root mean squared error 0.4381  
 Relative absolute error 88.0045 %  
 Root relative squared error 103.8636 %  
 Total Number of Instances 49

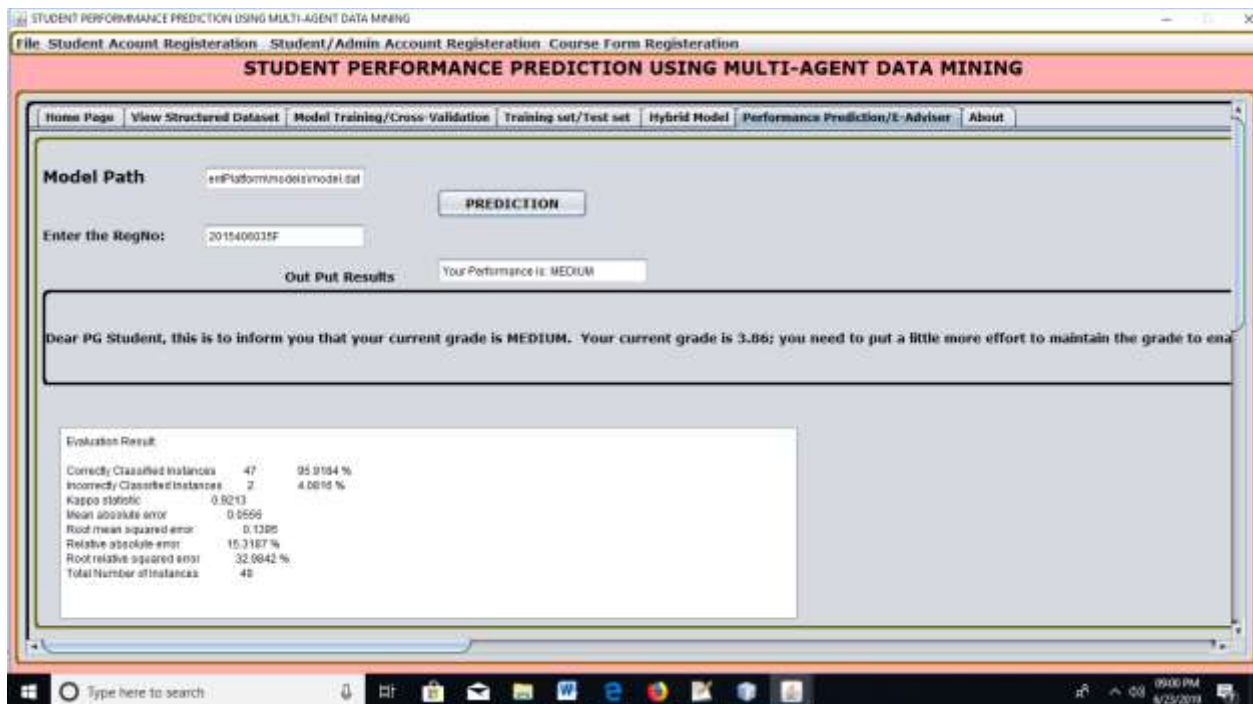
total\_instances : 49  
 correct pred. : 49  
 incorrect predictions : 0  
 accuracy : 89.38%

Type here to search 12:11 PM 7/10/2009

## Model Building and Evaluation Interface for K-NN



## Hybrid Model Building



## Student's Performance Prediction by Students' ID (MEDIUM Performance)

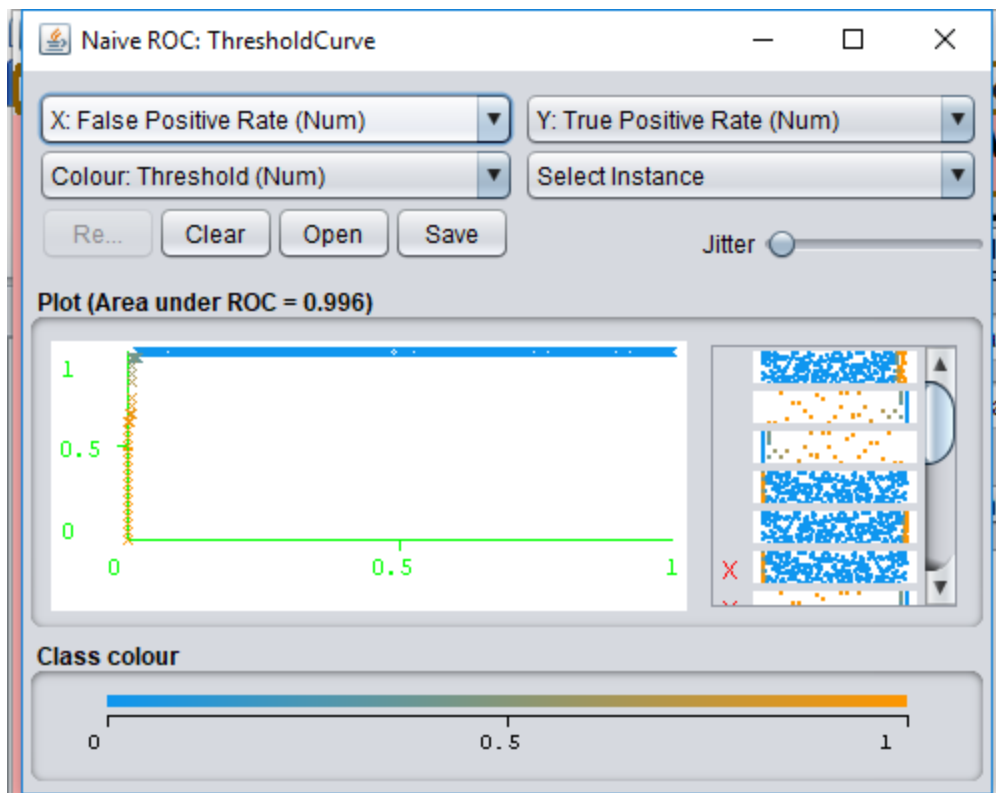
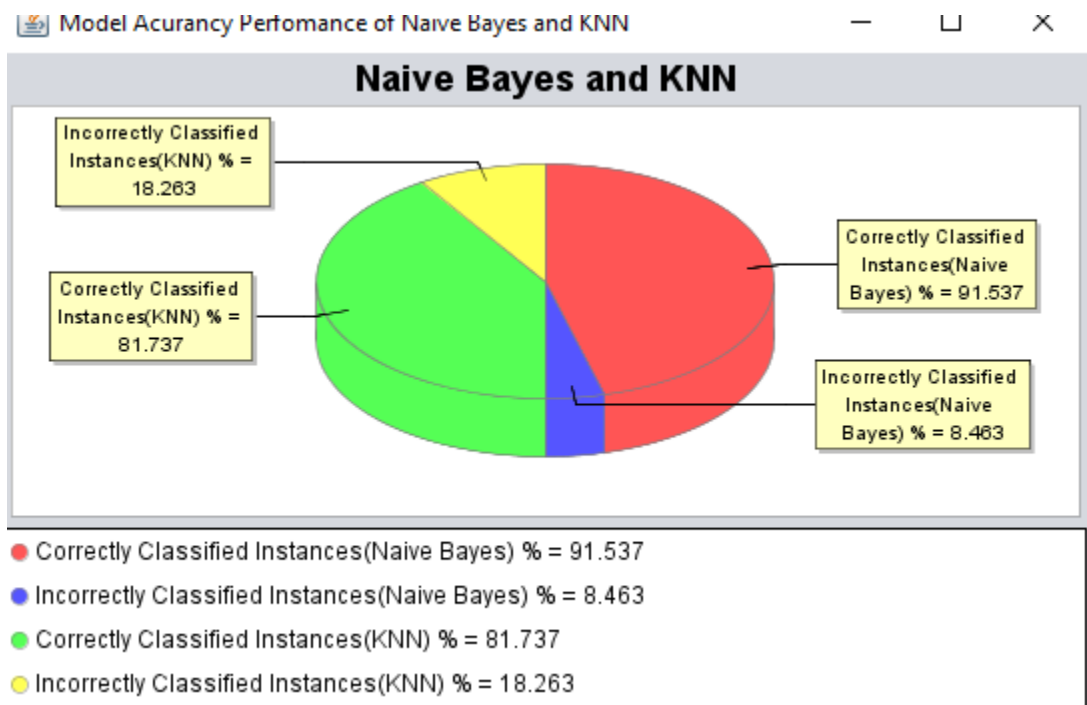
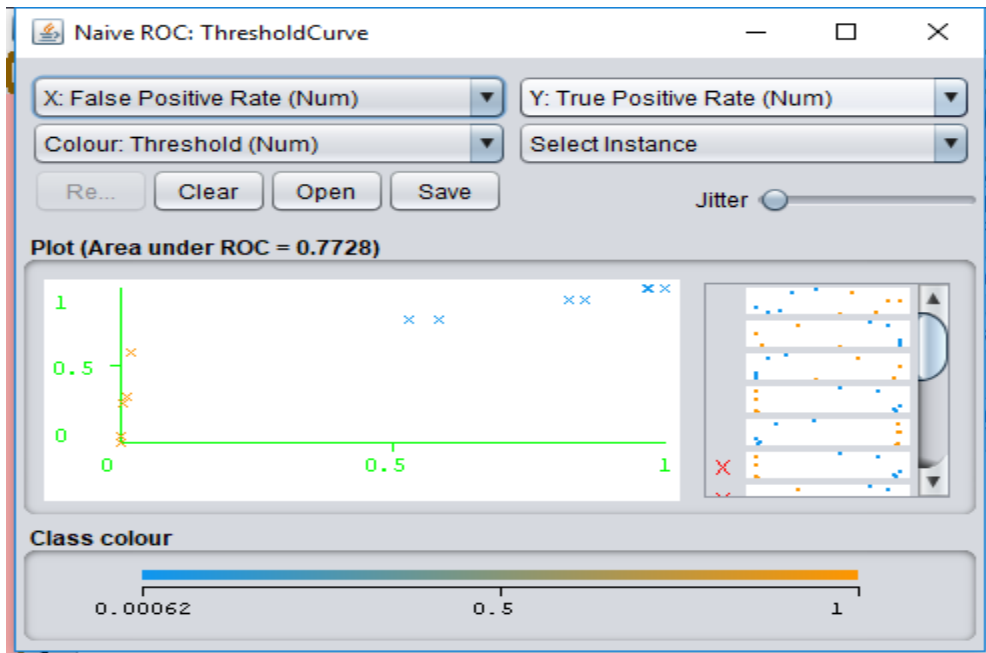


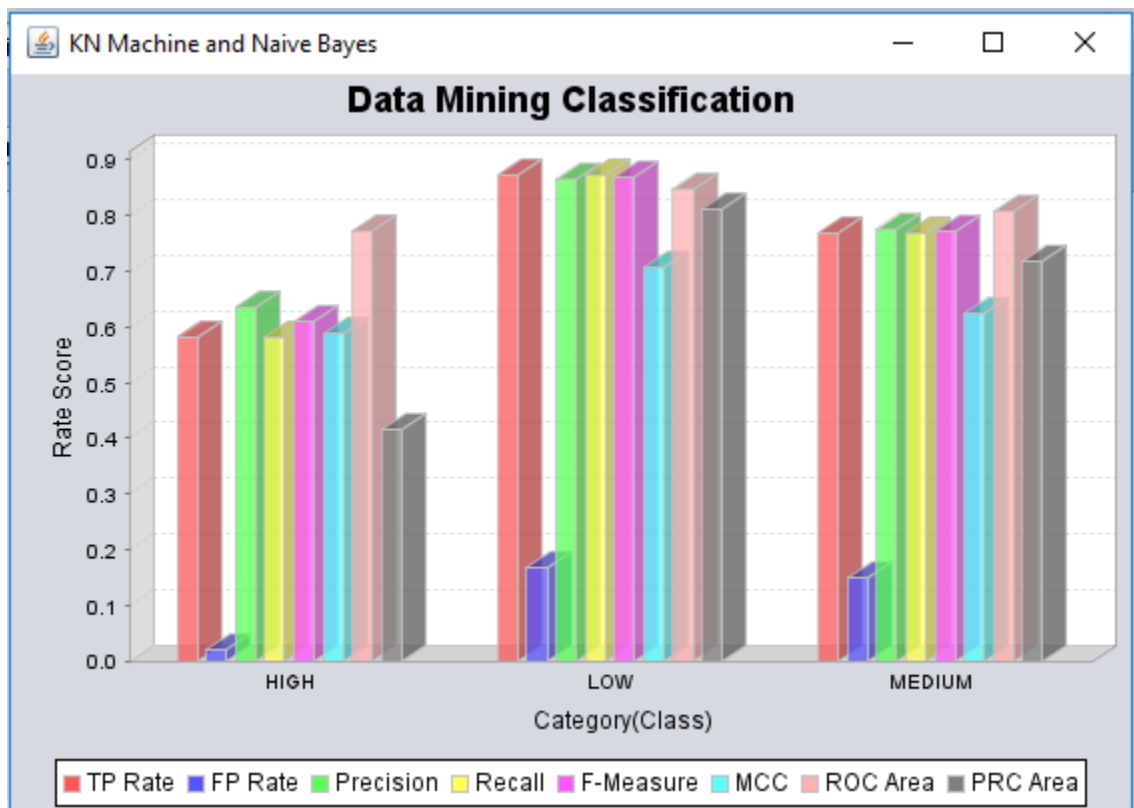
Figure ROC for Naive Bayes



Naïve Bayes and KNN



### ROC for K-NN



### K-NN and Naive Bayes



Bar Chart Hybrid model performance Evaluation

**STUDENT PERFORMANCE PREDICTION USING MULTI-AGENT DATA MINING**

Home Page | View Structured Dataset | Model Training/Cross-Validation | Training set/Test set | Hybrid Model | Performance Prediction/E-Adviser | **DEMOGRAPHIC FACTORS** | AWRF | SF | STU

**DEMOGRAPHIC FACTORS**

Student's Gender:  Male  Female

Marital Status:  Single  Married  Divorced

Program:  PGD  Masters  PHD

City of Residence:  Rural  Urban

Mode of Study:  Full Time  Part Time

Students' Age:  20-30  31-40  41-50  51 and above

Employment Status:  Full Time  Part Time  Unemployed

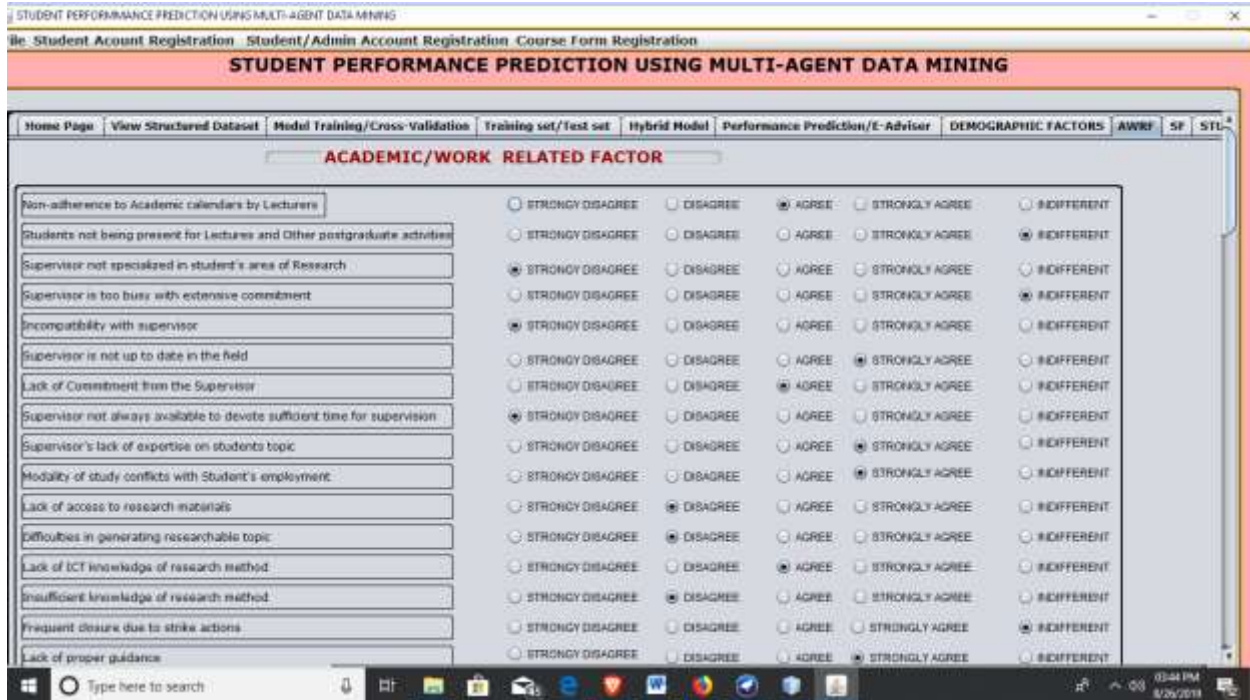
Family size:  3 Person  4 Person  5 Person and above

Job-Course-Relationship:  Closely related  Somewhat related  Not related  Unemployed

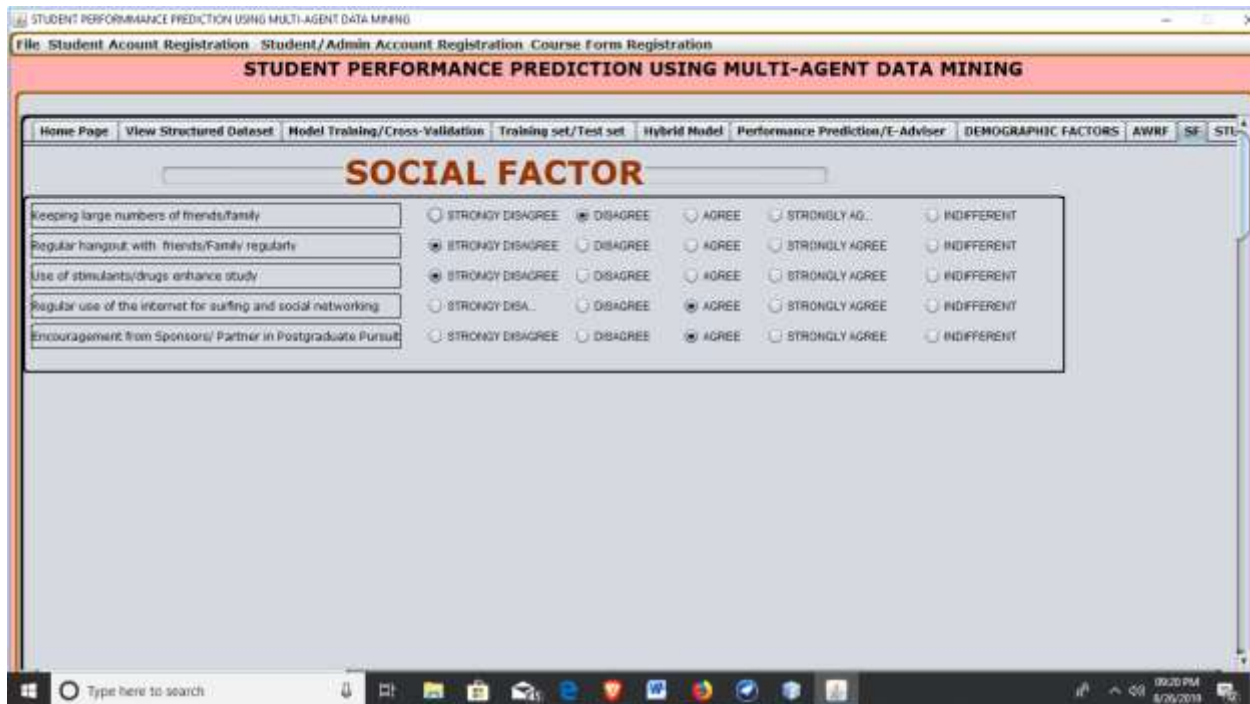
Sponsor:  Self sponsor  Parents/Guardian  Organization/Scholarship  TETFUND

Program Motivation:  Better Employment Prospects  To upgrade qualification  Pursuit of Knowledge  
 Personal Fulfillment  Lack of Employment

Demographic Factors



## Academic and Work Related Factors



## Social Factors



STUDENT PERFORMANCE PREDICTION USING MULTI-AGENT DATA MINING

File Student Account Registration Student/Admin Account Registration Course Form Registration

### STUDENT PERFORMANCE PREDICTION USING MULTI-AGENT DATA MINING

Home Page View Structured Dataset Model Training/Cross-Validation Training set/Test set Hybrid Model Performance Prediction/E-Adviser DEMOGRAPHIC FACTORS AWWF SF STU

Academic Course	Grade Scores
ACC811	59
ACC812	60
ACC815	61
ACC 817	76
ACC 819	63
ACC821	48
ACC826	70
ACC827	0
ACC829	0

2015409352F Search Engine

Enter Student RegNo Refresh

Type here to search

8:36 PM 8/24/2019

## Students Results



## Postgraduate Students Problems Questionnaire (PGSPQ)

The purpose of this questionnaire is to gather information of factors that affect Postgraduate Students' Academic performance.

Please, I urge you to fill in your responses and tick the options that are provided therein.

### A. DEMOGRAPHIC FACTORS:

1. Reg. No: \_\_\_\_\_
2. Sex: (1)F (2)M
3. Marital Status: (1)Single (2)Married (3) Divorced
4. What program are you currently running: (1) Postgraduate Diploma (2) Masters (3) Doctorate
5. City of Residence (1) Rural (2) Urban
6. Students' Mode of Study: (1)Full time (2) Part time
7. Students' Age (1)20-30yrs(2)31-40yrs(3)41-50yrs(4)50 and above
8. Are you currently employed full-time (35 hours per week or more) or part-time (1) Full Time (2) Part Time (3) Unemployed
9. Family size: (1)3persons (2)4persons (3)5persons and above
10. How closely related is your current job to the skills and concepts of your current course? (1) Closely related (2) Somewhat related (3) Not related (4) Unemployed
11. Who funds your Education (1) Self Sponsor (2) Parents/Guardian (3) Organization/Scholarship (4) TETFund
12. Sponsor's highest educational qualifications: (1) primary school (2)Junior Secondary School (3) Senior Secondary School (4) Senior Secondary School (5) higher Degree
13. Reasons for Pursuit of Postgraduate Studies
  - (1) Better employment prospect
  - (2) To upgrade qualification
  - (3) Pursuit of knowledge
  - (4) Personal fulfillment
  - (5) Lack of employment

Please rate your responses to the following questions using the scale below:

- (1) Strongly Disagree (2) Disagree (3) Agree (4) Strongly Agree (5) Indifferent

<b>B</b>	<b>ACADEMIC/WORK RELATED FACTOR</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
14	Non-adherence to Academic calendars by Lecturers					
15	Students not being present for Lectures and Other postgraduate activities					
17	Supervisor not specialized in student's area of Research					
18	Supervisor is too busy with extensive commitment					
19	Incompatibility with supervisor					
20	Supervisor is not up to date in the field					
21	Lack of Commitment from the Supervisor					

22	Supervisor not always available to devote sufficient time for supervision					
23	Supervisor's lack of expertise on students topic					
24	Modality of study conflicts with Student's employment					
25	Lack of access to research materials					
26	Difficulties in generating researchable topic					
27	Lack of ICT knowledge of research method					
28	Insufficient knowledge of research method					
29	Frequent closure due to strike actions					
30	Lack of proper guidance					
31	Problem of Funding					
32	Problem of accommodation					
33	Poor library facilities, Standard equipment and Laboratory					
34	Untimely submission of Postgraduate Semester results					

Please rate your responses to the following questions using the scale below:

(1) Strongly Disagree (2) Disagree (3) Agree (4) Strongly Agree (5) Indifferent

<b>C</b>	<b>SOCIAL FACTOR</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
35	Keeping large numbers of friends/family					
36	Regular hangout with friends/Family regularly					
37	Use of stimulants/drugs enhance study					
38	Regular use of the internet for surfing and social networking					
39	Encouragement from Sponsors/ Partner in Postgraduate Pursuit					