NETWORK INTRUSION DETECTION USING MACHINE LEARNING

Submitted in partial fulfillment of the requirements for
the award of
Bachelor of Engineering degree in Computer Science and Engineering

by

NithishBabu Gorantla (Roll. No. 17SCS0141)
Anudeep Indla (Roll. No. 17SCS0169)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC JEPPIAAR
NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600 119

MARCH - 2021

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with "A" grade by NAAC
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600119

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this project report is the bonafide work of **NithishBabu Gorantla (Roll. No. 17SCS0141) and Anudeep Indla (Roll. No. 17SCS0169)** who carried out the project entitled "**NETWORK INTRUSION DETECTION USING MACHINE LEARNING**" under my supervision from **August 2020** to **March 2021**.

Internal Guide
Dr. Joshila Grace, M.E., Ph.D.,

Head of the Department

Dr. S. Vigneshwari, M.E., Ph.D.,

Dr. L. Lakshmanan, M.E., Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner **External Examiner**

DECLARATION

NithishBabu Gorantla(37110237), Anudeep Indla(37110279) hereby declare that the Project Report entitled "**NETWORK INTRUSION DETECTION USING MACHINE LEARNING"** is done by me under the guidance of **Dr. Joshila Grace, M.E., Ph.D.,** Department of Computer Science and Engineering at Sathyabama Institute of Science and Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering**.**

DATE:

PLACE: CHENNAI **SIGNATURE OF THE CANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala, M.E., Ph.D., Dean**, School of Computing, **Dr. S. Vigneswari, M.E., Ph.D., and Dr. L. Lakshmanan, M.E., Ph.D., Heads of the Department** of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. Joshila Grace, M.E., Ph.D.,** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

The goal is to predict a Windows machine’s probability of getting infected by various families of malware, based on different properties of that machine. The telemetry data containing these properties and the machine infections was generated by combining heartbeat and threat reports collected by Microsoft's endpoint protection solution, Windows Defender. Each row in this dataset corresponds to a machine, uniquely identified by a Machine Identifier. Has Detections is the ground truth and indicates that Malware was detected on the machine. Using the information and labels in train.csv, you must predict the value for Has Detections for each machine in test.csv. The sampling methodology used to create this dataset was designed to meet certain business constraints, both in regards to user privacy as well as the time period during which the machine was running. Malware detection is inherently a time-series problem, but it is made complicated by the introduction of new machines, machines that come online and offline, machines that receive patches, machines that receive new operating systems, etc. While the dataset provided here has been roughly split by time, the complications and sampling requirements mentioned above may mean you may see imperfect agreement between your cross validation, public, and private scores! Additionally, this dataset is not representative of Microsoft customers machines in the wild it has been sampled to include a much larger proportion of malware machines.

TABLE OF CONTENTS

CONTENTS **PAGE NO**

CONTENTS **PAGE NO**

LIST OF FIGURES

1. INTRODUCTION

CHAPTER 1

INTRODUCTION

Introduction

For the past few years, network has played a significant role in communication. The computer network allows the computing network devices to exchange information among different systems and individuals. The services of various organizations, companies, colleges, universities are accessed throughout computer network. This leads to a massive growth in networking field. The accessibility of internet has acquired a lot of interest among individuals. In this context, security of information has become a great challenge in this modern area. The information or data that we would like to send is supposed to be secured in such a way that a third party should not take control over them. When we are talking about security, we have to keep three basic factors in our mind: Confidentiality, Integrity and availability. Confidentiality means privacy of information. It gives the formal users the right to access the system via internet. This can be performed suitably along with accountability services in order to identify the authorized individuals. The second key factor is integrity. The integrity service means exactness of information. It allows the users to have self- assurance that the information passed is acceptable and has not been changed by an illegal individual.

An Intrusion Detection System (IDS) is used to watch malicious activities over the network. It can sort the unfamiliar records as normal or attack class. First monitoring of the network traffic is done, and then the IDS sorts these network traffic records into either malicious class or regular class. It acts as an alarm system that reports when an illegal activity is detected. The exactness of the IDS depends upon detection rate. If the performance is high for the IDS, then the correctness of detection is also high. Some of the intrusion detection systems are marketed with the ability to stop attacks before they are successful. They are used to shield an association from attack. It is a relative concept that tries to identify a hacker when intrusion is attempted. Ideally, such a system will only alarm when a successful attack is made. Intrusion detection system is not a perfect solution to all attack types. The various goals that can be accomplished with an Intrusion Detection System are: The potential goals include the following:

- IDS detect attacks.
- IDS traces user activity from point of entry to

- IDS generate alerts when required.
- Detect errors in system configuration.
- Provides security of the system without the need of non – expert staff.
- IDS can detect when the system is under attack. Provides evidences for attack.

Purpose

We propose below methodology for solving the problem. Raw data collected would be pre-processed for missing data, anomalies and outliers. Then an algorithm would be trained on this data to create a model. This model would be used for forecasting the final results. ETL stands for Extract, Transform and load. It is a tool which is a combination of three functions. It is used to get data from one database and transform it into a suitable format. Data preprocessing is a data mining technique used to transform sample raw data into an understandable format. Real world collected data may be inconsistent, incomplete or contains an error and hence data preprocessing is required.

Scope

Range of Expertise Includes:

- Software Development Services
- Engineering Services
- Systems Integration
- Customer Relationship Management
- Product Development
- Electronic Commerce
- Consulting
- IT Outsourcing

We apply technology with innovation and responsibility to achieve two broad objectives:

- Effectively address the business issues our customers face today.
- Generate new opportunities that will help them stay ahead in the future.

This Approach Rests On:

- A strategy where we architect, integrate and manage technology services and solutions - we call it AIM for success.
- A robust offshore development methodology and reduced demand on customer resources.
- A focus on the use of reusable frameworks to provide cost and times benefits.

They combine the best people, processes and technology to achieve excellent results - consistency.
We offer customers the advantages of:

Speed:

They understand the importance of timing, of getting there before the competition. A rich portfolio of reusable, modular frameworks helps jump-start projects. Tried and tested methodology ensures that we follow a predictable, low - risk path to achieve results. Our track record is testimony to complex projects delivered within and evens before schedule.

Expertise:

Our teams combine cutting edge technology skills with rich domain expertise. What's equally important - they share a strong customer orientation that means they actually start by listening to the customer. They're focused on coming up with solutions that serve customer requirements today and anticipate future needs.

A Full Service Portfolio:

They offer customers the advantage of being able to Architect, integrate and manage technology services. This means that they can rely on one, fully accountable source instead of trying to integrate disparate multi vendor solutions.

Services:

Providing its services to companies which are in the field of production, quality control etc with their rich expertise and experience and information technology they are in best position to provide software solutions to distinct business requirements.

2. SYSTEM ANALYSIS

CHAPTER 2

SYSTEM ANALYSIS

Existing System

Network Intrusion Detection system is a mechanism that is used within the network to identify the malicious event. It uses K- Nearest Neighbor algorithm for intrusion Detection. The network traffic is monitored in the network that is in the sub- net. If an attack is observed it matches the traffic with the known attack list. Then an alert is passed to the administrator. Network Intrusion Detection System (NIDS) and Host Intrusion Detection System (HIDS) are the two most widely used systems for intrusion detection. NIDS is installed in router to identify the passage of network traffic. HIDS runs on an individual system. The functions of two IDSs are the same. HIDS also monitors the unauthorized activity. It takes a short review of the existing files in the system. Then it matches it with the old system files. If it finds an intrusion or changes in the system, then an alert is passed to the administrator. The intrusion can be detected as if a file is modified or deleted, then it means malicious activity is reported.

Design:

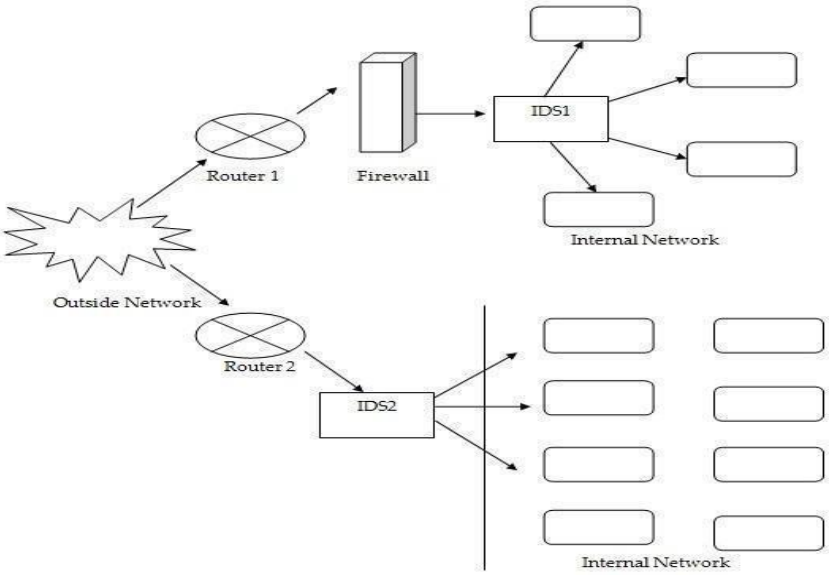


Fig 2.1: Existing Intrusion Detection System Design

K-Nearest neighbor:

K-Nearest neighbor is a lazy learner technique. This algorithm depends on learning by analogy. It is a supervised classification method. This classifier is used extensively for classification purpose. This classifier waits till the last minute prior to build some model on a specified tuple as compared to earlier classifiers. The training tuples are characterized in N-dimensional space in this classifier. This classification model looks for the k training tuples nearest to the indefinite sample in case of an indefinite tuple. Then, this classifier puts the sample in the closest class.

Disadvantages:

Results with less accurate which is less than 50% due to

1. **Does not work well with large dataset:** In large datasets, the cost of calculating the distance between the new point and each existing points is huge which degrades the performance of the algorithm.

2. **Does not work well with high dimensions:** The KNN algorithm doesn't work well with high dimensional data because with large number of dimensions, it becomes difficult for the algorithm to calculate the distance in each dimension.

3. **Need feature scaling:** We need to do feature scaling (standardization and normalization) before applying KNN algorithm to any dataset. If we don't do so, KNN may generate wrong predictions.

4. **Sensitive to noisy data, missing values and outliers**: KNN is sensitive to noise in the dataset. We need to manually impute missing values and remove outliers.

Proposed System

In Proposed system supervised method is used for detecting the Intrusion in the system. In order to increase the detection ability of IDS and prevent the service providers from attack, we propose an efficient ML based IDS using Light gradient boosting method and Random Forest algorithms. In order to overcome the problem of class imbalance, feature selection based on CFS-BA is used to determine a subset of the original features to eliminate irrelevant features. The detection framework of the proposed ML- Based consists of three stages including: feature selection, build and train the ensemble classifier and attack recognition.

Detailed information about the framework:

Feature Selection:

The aim of feature selection is to find a subset of the attributes from the original set which are representative enough for the data, and the attributions in the subset are highly relevant to the prediction. Feature selection approaches can be mainly categorized into wrapper, filter and embedded approaches. While filter approaches assess the relevance of the features from the dataset and the selection of the features is based on the statistics, the classification.The performance is used in wrapper approaches as a part of the feature subsets evaluation and selection processes. In contrast to wrapper approaches, embedded approaches are computationally less intensive than wrappers because they incorporate an interaction between feature selection and learning process. Modern intrusion detection datasets inevitably contain plenty of redundant and irrelevant attributes. Redundant and irrelevant attributes can lower the efficiency of data mining algorithms, causing uninterruptable results. Therefore, the first step in this study is to reduce the dimensionality and select the feature subset of the utilized dataset. In this paper, a hybrid approach by combining CFS with BA is proposed to optimize the efficiency of the feature selection process and enhance the accuracy of the classification.The main concept of this approach is to evaluate the relevance and the redundancy of the selected feature subset which is searched in the given search space for the optimal solution.

- **Correlation-based feature selection (CFS):** CFS is one of classical filter algorithms that choose features according to the result of the heuristic (correlation-based) assessment function. The preference of this function is to select subsets whose features are extraordinarily related with the class but uncorrelated with each other.

- **Bat algorithm (BA):** The original bat algorithm was developed by Xin-She Yang in 2010 . The main inspirations for these works were the echolocation behavior of micro bats.

Accuracy score for gradient boosting algorithm is 65.6%

Target variable:

The target variable of a dataset is the feature of a dataset about which you want to gain a deeper understanding. A supervised machine learning algorithm uses historical data to learn patterns and uncover relationships between other features of your dataset and the target.

Here we use target variable known as ‘Has detection’. Using this target variable we can know whether the system is intruded or not. The final output which we get is in the form of 0’s and 1’s.It means that 0 indicates that system has no detections,1 indicates that the system is having intrusion for detection.

Total result is obtained by using confusion matrix. The confusion matrix shows the ways in which our classification model is confused when it makes predictions.

Advantages:

Works on group of models make weaker models become stronger models and hence better accuracy hence it is called ensemble model.

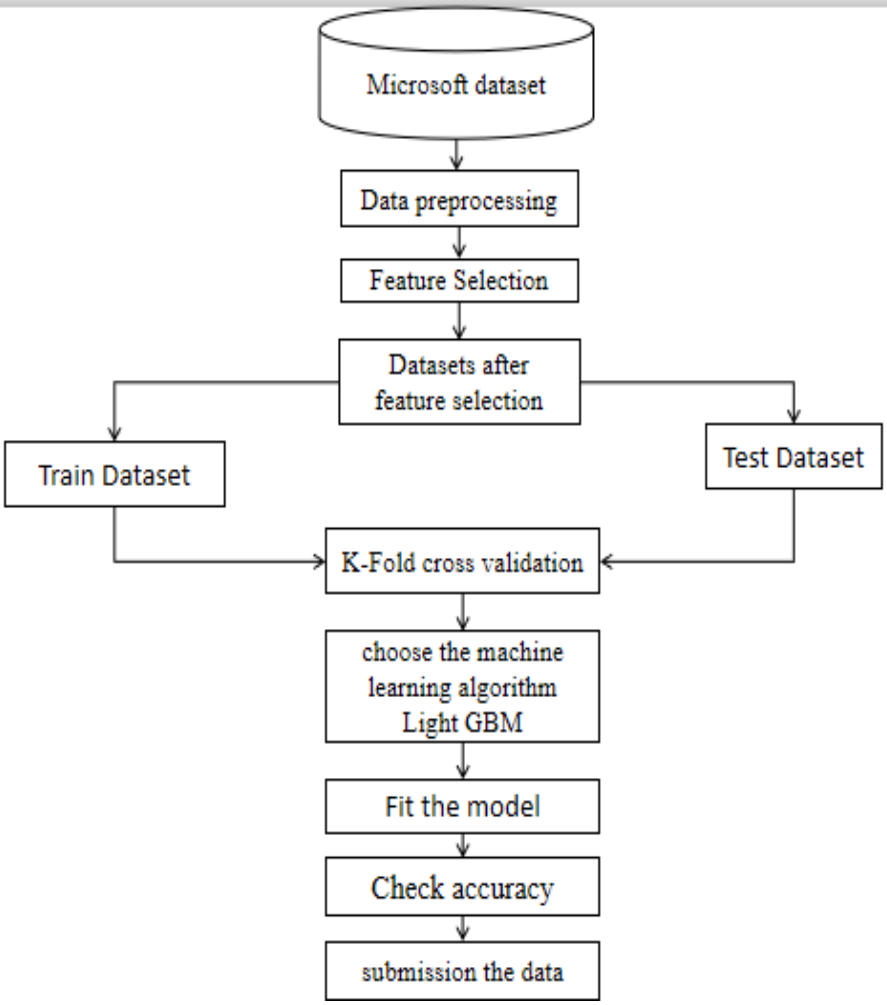


Fig 2.2: Proposed System Flow Chart

Model Description

Boosting is a method of converting weak learners into strong learners. In boosting, each new tree is a fit on a modified version of the original data set. Gradient Boosting trains many models in a gradual, additive and sequential manner. The gradient boosting algorithm (gbm) begins by training a decision tree in which each observation is assigned an equal weight. After evaluating the first tree, we increase the weights of those observations that are difficult to classify and lower the weights for those that are easy to classify. The second tree is therefore grown on this weighted data. Here, the idea is to improve upon the predictions of the first tree.

Accuracy score for gradient boosting algorithm is 64.0%

LightGBM:

LightGBM is a fast, distributed as well as high-performance gradient boosting (GBDT, GBRT, GBM or MART) framework that makes the use of a learning algorithm that is tree-based, and is used for ranking, classification as well as many other machine learning tasks.

Light GBM gaining popularity at an extreme level:

Day by day the size of data is increasing and it is becoming increasingly difficult for traditional data science algorithms to give faster results. Coming to Light GBM, it is prefixed as ‘Light’ because of its high speed. Light GBM, to its advantage, can handle the large size of data and takes lower memory to run.

One more reason why Light GBM is popular is that it focuses more on the accuracy of results. It also supports GPU learning and this is why data scientists are widely using LGBM for development of data science applications.

Difference of Light GBM from other tree-based algorithms:

While other algorithms grow trees horizontally, Light GBM grows tree vertically meaning that Light GBM grows tree leaf-wise while other algorithms grow level-wise. It in order to grow, will choose the leaf that has a max delta loss. When growing the same leaf, Leaf-wise algorithm can reduce more loss when compared to a level-wise algorithm.

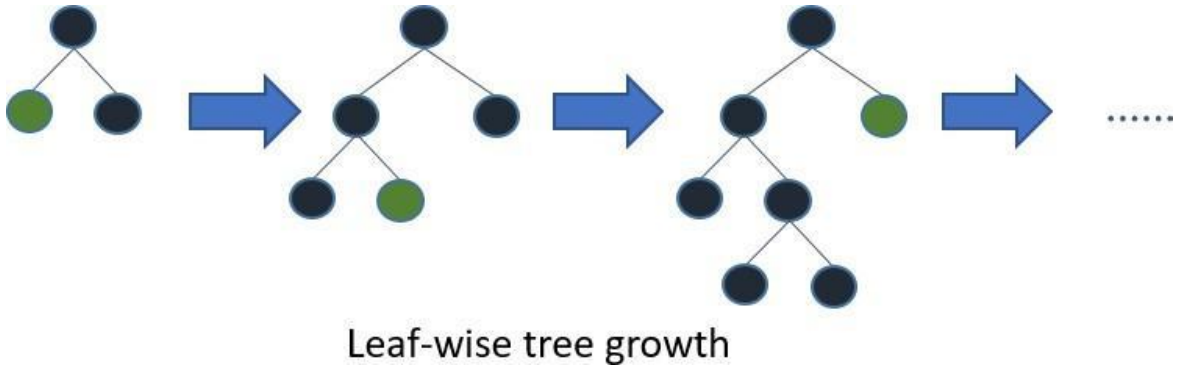


Fig 2.2.1: Light GBM working

Advantages:

LightGBM as we already know is a gradient boosting framework that makes the use of tree-based learning algorithms. It is designed with the following advantages in order to be distributed as well as efficient:

- Higher efficiency as well as faster training speed
- Usage of lower memory
- Better accuracy
- Supports Parallel and GPU learning
- Data of large-scale can be handled.

On the basis of all the experiments that have been performed on public datasets, it is shown that LightGBM with significantly lower memory consumption on both efficiency and accuracy, can outperform other existing boosting framework.

Adding on, the experiments also show that LightGBM by using multiple machines for training in specific settings can achieve a linear speed-up.

Study of the System

In the flexibility of uses the interface has been developed a graphics concepts in mind, associated through a browser interface. The GUI's at the top level has been categorized as follows

1. Administrative User Interface Design
2. The Operational and Generic User Interface Design

The administrative user interface concentrates on the consistent information that is practically, part of the organizational activities and which needs proper authentication for the data collection. The Interface helps the administration with all the transactional states like data insertion, data deletion, and data updating along with executive data search capabilities.

The operational and generic user interface helps the users upon the system in transactions through the existing data and required services. The operational user interface also helps the ordinary users in managing their own information helps the ordinary users in managing their own information in a customized manner as per the assisted flexibilities.

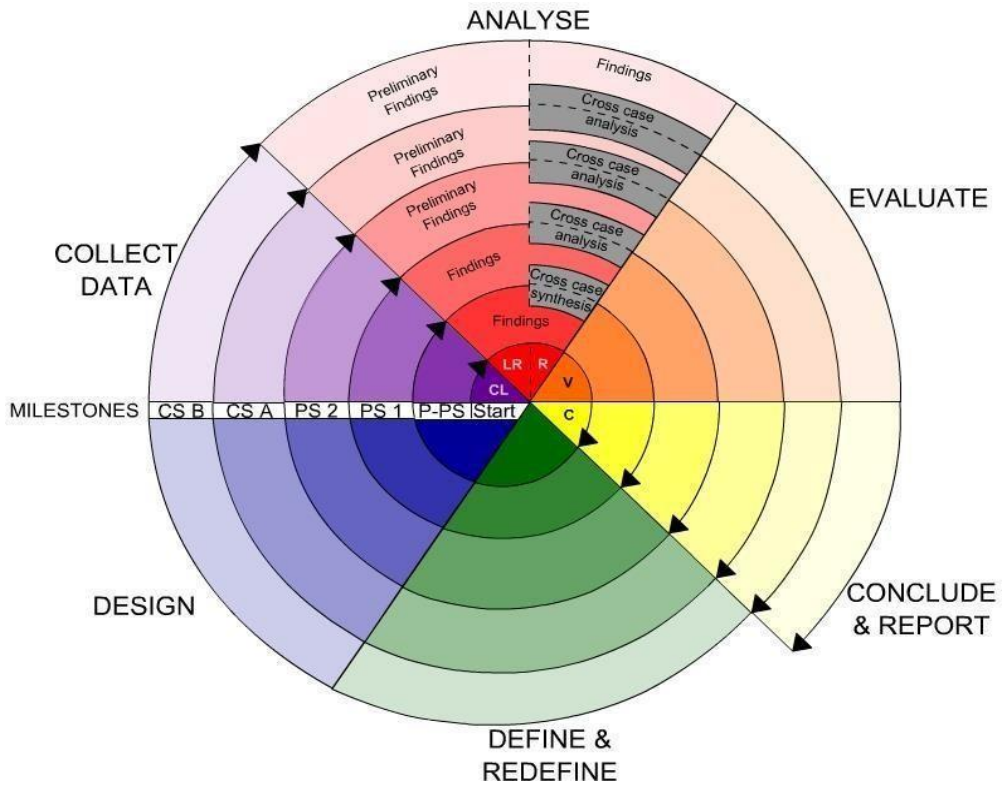


Fig 2.2.2: Model of the system

Fundamental Concepts on Domain

Cloud era:

Cloud era is revolutionizing enterprise data management by offering the first unified Platform for Big Data: The Enterprise Data Hub. Cloud era offers enterprises one place to store, process, and analyze all their data, empowering them to extend the value of existing investments while enabling fundamental new ways to derive value from their data.

Why do customers choose Cloud era:

Cloud era was the first commercial provider of python-related software and services and has the most customers with enterprise requirements, and the most experience supporting them, in the industry. Cloud era's combined offering of differentiated software (open and closed source), support, training, professional services, and indemnity brings customers the greatest business value, in the shortest amount of time, at the lowest TCO.

Data Mining

There is a huge amount of data available in the Information Industry. This data is of no use until it is converted into useful information. It is necessary to analyze this huge amount of data and extract useful information from it. Extraction of information is not the only process we need to perform; data mining also involves other processes such as Data Cleaning, Data Integration, Data Transformation, Data Mining, Pattern Evaluation and Data Presentation. Once all these processes are over, we would be able to use this information in many applications such as Fraud Detection, Market Analysis, Production Control, Science Exploration, etc.

Data Mining:

Data Mining is defined as extracting information from huge sets of data. In other words, we can say that data mining is the procedure of mining knowledge from data. The information or knowledge extracted so can be used for any of the following applications:

Data Mining Applications:

Data mining is highly useful in the following domains: Market Analysis and Management Corporate Analysis & Risk Management Fraud Detection Apart from these, data mining can also be

used in the areas of production control, customer retention, science exploration, sports, astrology, and Internet Web Surf-Aid

Machine Learning

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves.

Uses:

Consider how you would write a spam filter using traditional programming techniques

1. First you would look at what spam typically looks like. You might notice that some words or phrases (such as “credit card,” “free,” and “amazing”) tend to come up a lot in the subject. Perhaps you would also notice a few other patterns in the sender’s name, the email’s body, and so on.

2. You would write a detection algorithm for each of the patterns that you noticed, and your program would flag emails as spam if a number of these patterns are detected.

3. You would test your program, and repeat steps 1 and 2 until it is good enough.

There are four types of machine learning:

1. Supervised Learning

In supervised learning, the training data you feed to the algorithm includes the desired solutions, called labels.

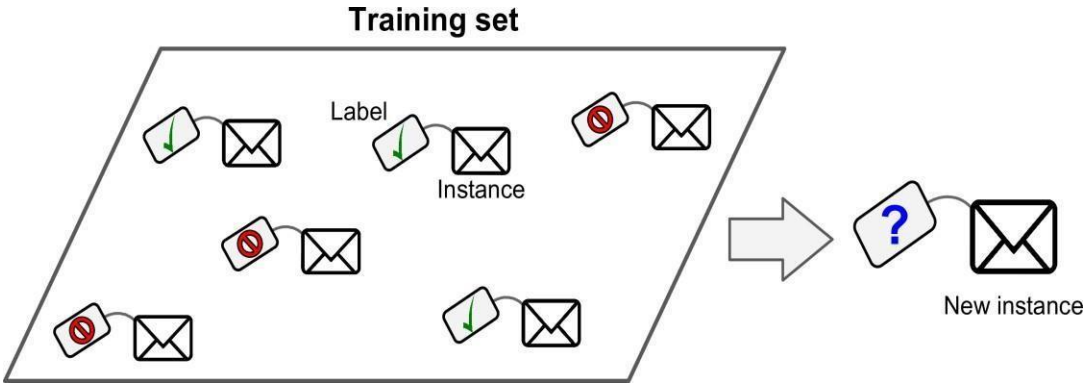


Fig 2.5.1: Supervised Learning

A typical supervised learning task is classification. The spam filter is a good example of this: it is trained with many example emails along with their class (spam or ham), and it must learn how to classify new emails.

Here are some of the most important supervised learning algorithms:

• k-Nearest Neighbors

• Linear Regression

• Logistic Regression

• Support Vector Machines (SVMs)

• Decision Trees and Random Forests

• Light GBM

2. Unsupervised Learning

In unsupervised learning, as you might guess, the training data is unlabeled .The system tries to learn without a teacher.

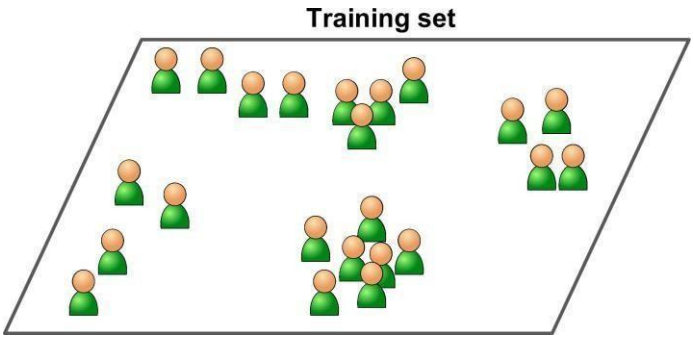


Fig.2.5.2: An unlabeled training set for unsupervised learning

Here are some of the most important unsupervised learning algorithms

• Clustering

- K-Means
- DBSCAN
- Hierarchical Cluster Analysis (HCA)

3. Semi-Supervised Learning

4. Reinforcement Learning

Data Cleaning:

Most Machine Learning algorithms cannot work with missing features, so let's create a few functions to take care of them. You noticed earlier that the total bedrooms attribute has some missing values, so let's fix this. You have three options:

- Get rid of the corresponding districts.
- Get rid of the whole attribute.
- Set the values to some value (zero, the mean, the median, etc.).

Methods:

- Feature Scaling
- Scaling and Standardization

Testing and Validating:

The only way to know how well a model will generalize to new cases is to actually try it out on new cases. One way to do that is to put your model in production and monitor how well it performs. This works well, but if your model is horribly bad, yourusers will complain—not the best idea.

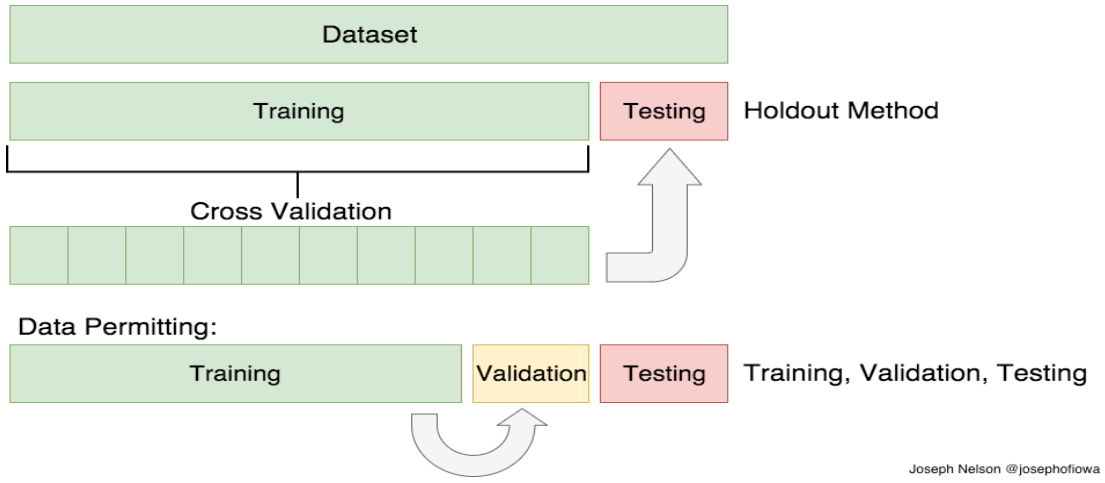


Fig 2.5.3: Cross Validation

A better option is to split your data into two sets: the training set and the test set. As these names imply, you train your model using the training set, and you test it using the test set. The error rate on new cases is called the generalization error (or out-of sample error), and by evaluating your model on the test set, you get an estimate of this error. This value tells you how well your model will perform on instances it has never seen before. If the training error is low (i.e., your model makes few mistakes on the training set) but the generalization error is high, it means that your model is over fitting the training data.

Confusion Matrix in Machine Learning:

In the field of machine learning and specifically the problem of statistical classification, a confusion matrix, also known as an error matrix. A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known. It allows the visualization of the performance of an algorithm. It allows easy identification of confusion between classes e.g. one class is commonly mislabeled as the other. Most performance measures are computed from the confusion matrix.

Confusion Matrix:

A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key to the confusion matrix. The confusion matrix shows the ways in which your classification model is confused when it makes predictions. It gives us insight not only into the errors being made by a classifier but more importantly the types of errors that are being made.

Here,

- Class 1 : Positive
- Class 2 : Negative

Classification Rate/Accuracy:

Classification Rate or Accuracy is given by the relation:

$$\textbf{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

However, there are problems with accuracy. It assumes equal costs for both kinds of errors. A 99% accuracy can be excellent, good, mediocre, poor or terrible depending upon the problem.

Recall:

$$Recall = \frac{TP}{TP + FN}$$

Recall can be defined as the ratio of the total number of correctly classified positive examples divide to the total number of positive examples. High Recall indicates the class is correctly recognized (a small number of FN).

Precision:

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

To get the value of precision we divide the total number of correctly classified positive examples by the total number of predicted positive examples. High Precision indicates an example labelled as positive is indeed positive (a small number of FP).

High recall, low precision: This means that most of the positive examples are correctly recognized (low FN) but there are a lot of false positives.

Low recall, high precision: This shows that we miss a lot of positive examples (high FN) but those we predict as positive are indeed positive (low FP).

ROC Curve:

The receiver operating characteristic (ROC) curve is another common tool used with binary classifiers. It is very similar to the precision/recall curve, but instead of plotting precision versus recall, the ROC curve plots the true positive rate (another name for recall) against the false positive rate. The FPR is the ratio of negative instances that are incorrectly classified as positive. It is equal to one minus the true negative rate, which is the ratio of negative instances that are correctly classified as negative. The TNR is also called specificity. Hence the ROC curve plots sensitivity (recall) versus 1 – specificity. To plot the ROC curve, you first need to compute the TPR and FPR for various threshold values, using the roc_curve() function:

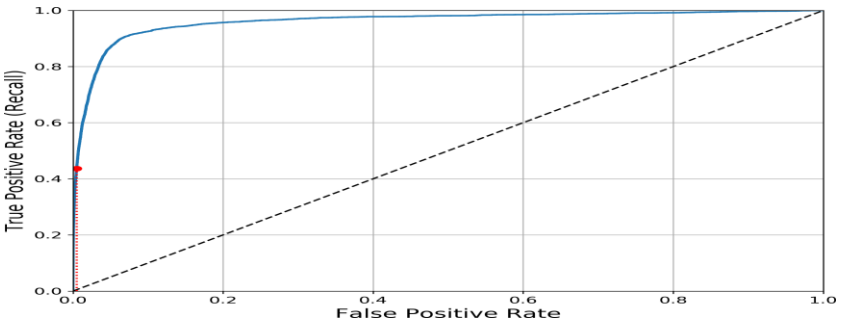


Fig 2.5.4: ROC curve

Once again there is a tradeoff: the higher the recall (TPR), the more false positives (FPR) the classifier produces. The dotted line represents the ROC curve of a purely random classifier; a good classifier stays as far away from that line as possible (toward the top-left corner). One way to compare classifiers is to measure the area under the curve (AUC). A perfect classifier will have a ROC AUC equal to 1, whereas a purely random classifier will have a ROC AUC equal to 0.5. Scikit- Learn provides a function to compute the ROC.

Feasibility Study:

A Feasibility Study is a preliminary study undertaken before the real work of a project starts to ascertain the likely hood of the projects success. It is an analysis of possible alternative solutions to a problem and a recommendation on the best alternative. Economic Feasibility Technical Feasibility Operational Feasibility Economic Feasibility It is defined as the process of assessing the benefits and costs associated with the development of project. A proposed system, which is both operationally and technically feasible, must be a good investment for the organization. With the proposed system the users are greatly benefited as the users can be able to detect the fakeness from the real news and are aware of most real and most fake news published in the recent years. This proposed system does not need any additional software and high system configuration. Hence the proposed system is economically feasible.

Technical Feasibility:

The technical feasibility in the proposed system deals with the technology used in the system. It deals with the hardware and software used in the system whether they are of latest technology or not. It happens that after a system is prepared a new technology arises and the user wants the system based on that technology. Technical Feasibility The technical feasibility infers whether the proposed system can be developed considering the technical issues like availability of the necessary technology, technical capacity, adequate response and extensibility. The project is decided to build using Python. Jupiter Note Book is designed for use in distributed environment of the internet and for the professional programmer it is easy to learn and use effectively. As the developing organization has all the resources available to build the system therefore the proposed system is technically feasible.

Economical Feasibility:

Economic analysis is the most frequently used method for evaluating theeffectiveness of a new system. More commonly known as cost/benefit analysis.

Operational Feasibility:

The project has been developed in such a way that it becomes very easy even for aperson with little computer knowledge to operate it. This software is very user friendly anddoes require any technical person to operate. Thus, the project is even operationally feasible. . Operational feasibility is defined as the process of assessing the degree to which a proposed system solves business problems or takes advantage of business opportunities.

3. REQUIREMENT SPECIFICATION

CHAPTER 3
REQUIREMENT SPECIFICATION

System Requirements

Hardware Requirements:

- **System** **:** Intel I-3, 5, 7 Processor.
- **Hard Disk** **:** 500 GB.
- **Floppy Drive :** 1.44 MB.
- **Monitor** : 14 Colour Monitor.
- **Mouse** **:** Optical Mouse.
- **RAM** **:** 2 GB.

Software Requirements:

- **Operating system** **:** Windows 7,8,10 Ultimate, Linux, Mac.
- **Front-End** **:** Python.
- **Coding Language** **:** Python.
- **Software Environment** **:** Anaconda.

Non-Functional Requirements

Output Design:

Outputs from computer systems are required primarily to communicate the results of processing to users. They are also used to provides a permanent copy of the results for later consultation. The various types of outputs in general are:

- External Outputs, whose destination is outside the organization
- Internal Outputs whose destination is within organization and they are the

- User’s main interface with the computer.
- Operational outputs whose use is purely within the computer department.
- Interface outputs, which involve the user in communicating directly.

Output Definition:

The Outputs should be defined in terms of following points

- Type of the output
- Content of the output
- Format of the output
- Location of the output
- Frequency of the output
- Volume of the output
- Sequence of the output

It is not always desirable to print or display data as it is held on a computer. It should be decided as which form of the output is the most suitable.

Input Design:

Input design is a part of overall system design. The main objective during the input design is as given below:

- To produce a cost-effective method of input.
- To achieve the highest possible level of accuracy.
- To ensure that the input is acceptable and understood by the user.

Input Stages:

The main input stages can be listed as below:

- Data recording
- Data transcription
- Data conversion
- Data verification
- Data control
- Data transmission
- Data validation

- Data correction

Input Types:

It is necessary to determine the various types of inputs. Inputs can be categorized as follows:

- External inputs, which are prime inputs for the system.
- Internal inputs, which are user communications with the system.
- Operational, which are computer department's communications to the system?
- Interactive, which are inputs entered during a dialogue.

Error Avoidance:

At this stage care is to be taken to ensure that input data remains accurate form the stage at which it is recorded up to the stage in which the data is accepted by the system. This can be achieved only by means of careful control each time the data is handled.

Error Detection:

Even though every effort is make to avoid the occurrence of errors, still a small proportion of Errors is always likely to occur, these types of errors can be discovered by using validations to check the input data.

Data Validation:

Procedures are designed to detect errors in data at a lower level of detail. Data validations have been included in the system in almost every area where there is a possibility for the user to commit errors. The system will not accept invalid data. Whenever an invalid data is keyed in, the system immediately prompts the user and the user has to again key in t he data and the system will accept the data only if the data is correct. Validations have been included where necessary.

The system is designed to be a user friendly one. In other words the system has been designed to communicate effectively with the user. The system has been designed with popup menus.

Functional requirements

Outputs from computer systems are required primarily to communicate the results of processing to users. They are also used to provide a permanent copy of the results for later consultation. The various types of outputs in general are:

- External Outputs, whose destination is outside the organization,.
- Internal Outputs whose destination is within organization and they are the
- User's main interface with the computer.
- Operational outputs whose use is purely within the computer department.
- Interface outputs, which involve the user in communicating directly.
- Understanding user's preferences, expertise level and his business requirements through a friendly questionnaire.
- Input data can be in four different forms - Relational DB, text files, .xls and xml files. For testing and demo you can choose data from any domain. User B provide input.

4. LANGUAGES IMPLEMENTATION

CHAPTER 4

LANGUAGES IMPLEMENTATION

Introduction to Python

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991. The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.

Python concepts

If you're not interested in the haws and whys of Python, feel free to skip to the next chapter. In this chapter I will try to explain to the reader why I think Python is one of the best languages available and why it's a great one to start programming with.

- Open source general-purpose language.
- Object Oriented, Procedural, Functional
- Easy to interface with C/Object/Java/Fortran
- Easy to interface with C++ (via SWIG)
- Great interactive environment

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted** − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive** − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** − Python supports Object−Oriented style or technique of programming that encapsulates code within objects.

- **Python is a Beginner's Language** – Python is a great language for the beginner– level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

PythonFeatures

Python's features include –

- **Easy-to-learn** – Python has few keywords, simple structure, and a clearly definedsyntax. This allows the student to pick up the language quickly.
- **Easy-to-read** – Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** – Python's source code is fairly easy–to–maintain.
- **A broad standard library** – Python's bulk of the library is very portable and cross– platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** – Python has support for an interactive mode which allowsinteractive testing and debugging of snippets of code.
- **Portable** – Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- **Extendable** – you can add low–level modules to the Python interpreter. Thesemodules enable programmers to add to or customize their tools to be more efficient.
- **Databases** – Python provides interfaces to all major commercial databases.
- **GUI Programming** – Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.

Python libraries

- **Request** The most famous http library written by kenneth reitz. It’s a must have forevery python developer.
- **Scrapy** If you are involved in webscraping then this is a must have library for you.After using this library you won’t use any other.
- **wxPython** A gui toolkit for python. I have primarily used it in place of tkinter. Youwill really love it.

- **Pillow** A friendly fork of PIL (Python Imaging Library). It is more user friendly than PIL and is a must have for anyone who works with images.
- **SQLAlchemy** A database library. Many love it and many hate it. The choice is yours.
- BeautifulSoup. I know it's slow but this xml and html parsing library is very useful for beginners.
- **Twisted** The most important tool for any network application developer. It has a very beautiful api and is used by a lot of famous python developers.
- **NumPy** How can we leave this very important library? It provides some advance math functionalities to python.
- **SciPy** When we talk about NumPy then we have to talk about scipy. It is a library of algorithms and mathematical tools for python and has caused many scientists to switch from ruby to python.
- **Matplotlib** A numerical plotting library. It is very useful for any data scientist or any data analyzer.
- **Pygame** Which developer does not like to play games and develop them? This library will help you achieve your goal of 2d game development.
- **Pyglet** A 3d animation and game creation engine. This is the engine in which the famous python port of mine craft was made
- **pyQT** A GUI toolkit for python. It is my second choice after wxpython for developing GUI's for my python scripts.
- **PyGtk** Another python GUI library. It is the same library in which the famous Bittorrent client is created.
- **Scapy** A packet sniffer and analyzer for python made in python.

Python modules:

Python allows us to store our code in files (also called modules). This is very usefulfor more serious programming, where we do not want to retype a long function definition

from the very beginning just to change one mistake. In doing this, we are essentially defining our own modules, just like the modules defined already in the Python library.
To support this, Python has a way to put definitions in a file and use them in a script or in an interactive instance of the interpreter. Such a file is called a module definitions from a module can be imported into other modules or into the main module.

Python Database Communication

Connector/Python provides a connect () call used to establish connections to the MySQL server. The following sections describe the permitted arguments for connect () and describe how to use option files that supply additional arguments. A database is an organizedcollection of data. The data are typically organized to model aspects of reality in a way that supports processes requiring this information. The term "database" can both refer to the datathem selves or to the database management system.

The Database management system is a software application for the interaction between users database itself. Databases are popular for many applications, especially for use with w applications or customer-oriented programs Users don't have to be human users. They can be other programs and applications as well. We will learn how Python or better a Python program can interact as a user of a SQLdatabase. This is an introduction into using SQLite and MySQL from Python. The Python standard for database interfaces is the Python DB- API, which is used by Python's database interfaces.
The DB-API has been defined as a common interface, which can be used to access relational databases. In other words, the code in Python for communicating with a database should be the same, regardless of the database and the database module used. Even though we use lots of SQL examples, this is not an introduction into SQL but a tutorial on the Python interface. SQLite is a simple relational database system, which saves its data in regular data files or even in the internal memory of the computer, i.e. the RAM. It was developed for embedded applications, like Mozilla- Firefox (Bookmarks), Symbian OS or Android. SQLITE is "quite" fast, even though it uses a simple file. It can be used for large databases as well. If you wantto use SQLite, you have to import the module sqlite3. To use a database, you have to create first a Connection object. The connection object will represent the database. The argument of connection - in the following example "companys.db" - functions both as the name of the file, where the data will be stored, and as the name of the database. If a file with this name

exists, it will be opened. It has to be a SQLite database file of course! In the following example, we will open a database called company. MySQL Connector/Python enables Python programs to access MySQL databases, using an API that is compliant with the Python Database API Specification v2.0 (PEP 249). It is written in pure Python and does not have any dependencies except for the Python Standard Library. For notes detailing the changes in each release of Connector/Python, see MySQL Connector/Python Release Notes.

MySQL Connector/Python includes support for:

- Almost all features provided by MySQL Server up to and including MySQL Server version 5.7.
- Converting parameter values back and forth between Python and MySQL data types, for example Python date time and MySQL DATETIME. You can turn automatic conversion on for convenience, or off for optimal performance.
- All MySQL extensions to standard SQL syntax.
- Protocol compression, which enables compressing the data stream between the client and server.
- Connections using TCP/IP sockets and on Unix using Unix sockets.
- Secure TCP/IP connections using SSL.
- Self-contained driver. Connector/Python does not require the MySQL client library or any Python modules outside the standard library.

NumPy

- **NumPy** is a general-purpose array-processing package. It provides a high performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python.
- Besides its obvious scientific uses, NumPy can also be used as an efficient multi- dimensional container of generic data. Array in NumPy is a table of elements (usually numbers), all of the same type, indexed by a tuple of positive integers.
- In NumPy, number of dimensions of the array is called rank of the array. A tuple of

integers giving the size of the array along each dimension is known as shape of the array. An array class in NumPy is called as NDarray. Elements in NumPy arrays are accessed by using square brackets and can be initialized by using nested Python Lists.

- NumPy's main object is the homogeneous multidimensional array. It is a table of elements (usually numbers), all of the same type, indexed by a tuple of positive integers. In NumPy dimensions are called *axes***.** The number of axes is rank**..**

Matplotlib:

High quality plotting library.

Big Data

Data

The quantities, characters, or symbols on which operations are performed by a computer, which May be stored and transmitted in the form of electrical signals and recorded on magnetic, optical, or mechanical recording media.

Big Data

Big Data is also data but with a huge size. Big Data is a term used to describe a collection of data that is huge in size and yet growing exponentially with time. In shortsuch data is so large and complex that none of the traditional data management tools areable to store it or process it efficiently.

Examples of Big Data

- The New York Stock Exchange generates about *one terabyte* of new trade data per day.

  Social Media.

- The statistic shows that *500+terabytes* of new data get ingested into the databases of social media site Facebook, every day. This data is mainly generated in terms of photo and video uploads, message exchanges, putting comments etc.

A single **Jet engine** can generate 10+terabytes of data in 30 minutes of flight time. With many

thousand flights per day, generation of data reaches up to many Peta

bytes. Types of Big Data

Big Data' could be found in three forms:

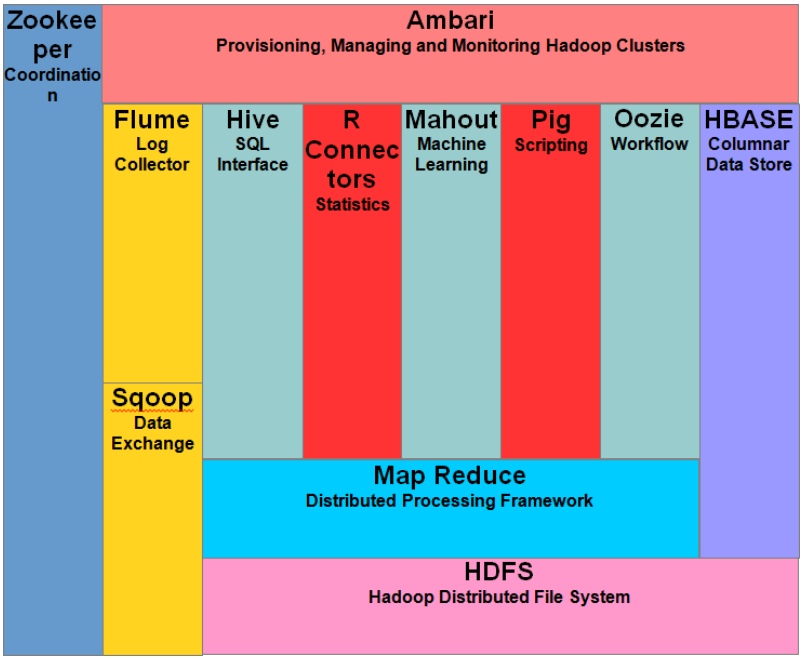
1. **Structured**
2. **Unstructured**
3. **Semi-structured**

Hadoop

Apache Hadoop is an open source software framework used to develop data processing applications which are executed in a distributed computing environment. Applications built using HADOOP are run on large data sets distributed across clusters of commodity computers. Commoditycomputers are cheap and widely available. These are mainly useful for achieving greater computational power at low cost. Similar to data residing in a local file system of a personal computer system, in Hadoop, data resides in a distributed file system which is called as a **Hadoop Distributed File system**. The processing model is based on **'Data Locality'** concept wherein computational logic is sent to cluster nodes (server) containing data. This computational logic is nothing, but a compiled version of a program written in a high-level language such as Java. Such a program, processes data stored in Hadoop HDFS.

Apache Hadoop consists of two sub-projects:

- Hadoop Map Reduce
- Hadoop Distributed File System



Hadoop Component Diagram

1. Hadoop Map Reduce: Map Reduce is a computational model and software framework for writing applications which are run on Hadoop. These Map Reduce programs are capable of processing enormous data in parallel on large clusters of computation nodes.
2. HDFS (Hadoop Distributed File System): HDFS takes care of the storage part of Hadoop applications. Map Reduce applications consume data from HDFS. HDFS creates multiple replicas of data blocks and distributes them on compute nodes in a cluster. This distribution enables reliable and extremely rapid computations.

Although Hadoop is best known for Map Reduce and its distributed file system- HDFS, the term is also used for a family of related projects that fall under the umbrella of distributed computing and large-scale data processing. Other Hadoop-related projects at Apache include are Hive, HBase, Mahout, Sqoop, Flume, and Zookeeper.

Jupiter Notebook

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data

visualization, machine learning.

Testing Code

- As indicated above, code is usually developed in a file using an editor.
- To test the code, import it into a Python session and try to run it.
- Usually there is an error, so you go back to the file, make a correction, and test again.
- This process is repeated until you are satisfied that the code works.
- The entire process is known as the development cycle.
- There are two types of errors that you will encounter. Syntax errors occur when the form of some command is invalid.
- This happens when you make typing errors such as misspellings, or call something by the wrong name, and for many other reasons. Python will always give an error message for a syntax error.

Sample Code

```
def convert_types(df):
    # Convert data types to reduce memory
    for c in df:
        col_type = str(df[c].dtypes)
        numerics = ['int16', 'int32', 'int64', 'float16', 'float32', 'float64']

        # Convert objects to category
        if col_type == 'object':
            df[c] = df[c].astype('category')

        # numerics
        elif col_type in numerics:
            c_min = df[c].min()
            c_max = df[c].max()
            if col_type[:3] == 'int':
                if c_min > np.iinfo(np.int8).min and c_max < np.iinfo(np.int8).max:
                    df[c] = df[c].astype(np.int8)
                elif c_min > np.iinfo(np.int16).min and c_max < np.iinfo(np.int16).max:
                    df[c] = df[c].astype(np.int16)
                elif c_min > np.iinfo(np.int32).min and c_max < np.iinfo(np.int32).max:
                    df[c] = df[c].astype(np.int32)
                elif c_min > np.iinfo(np.int64).min and c_max < np.iinfo(np.int64).max:
                    df[c] = df[c].astype(np.int64)
            else:
                if c_min > np.finfo(np.float16).min and c_max < np.finfo(np.float16).max:
                    df[c] = df[c].astype(np.float16)
                elif c_min > np.finfo(np.float32).min and c_max < np.finfo(np.float32).max:
                    df[c] = df[c].astype(np.float32)
                else:
                    df[c] = df[c].astype(np.float64)

    return df
```

Fig4.7.1: optimizing the features

```
numerical_features = excel_table[excel_table['FeatureType']=='Numeric']['Feature'].reset_index(drop=True)
categorical_features = excel_table[excel_table['FeatureType']=='Category']['Feature'].reset_index(drop=True)
binary_features = excel_table[excel_table['FeatureType']=='Boolean']['Feature'].reset_index(drop=True)
```

Fig4.7.2: Categorizing the features

Fig4.7.3: Correlation between the Attributes

```
In [22]: model.fit(X_train, y_train, categorical_feature=categorical_feature)
         # time: 1h 52min 39s
```

```
C:\Users\Test\Anaconda3\lib\site-packages\lightgbm\basic.py:1295: UserWarning: categorical_feature in Dataset is overridden.
New categorical_feature is ['Census_ActivationChannel', 'Census_ChassisTypeName', 'Census_DeviceFamily', 'Census_FlightRing',
'Census_GenuineStateName', 'Census_HasOpticalDiskDrive', 'Census_InternalPrimaryDiagonalDisplaySizeInInches', 'Census_IsAlwaysO
nAlwaysConnectedCapable', 'Census_IsPenCapable', 'Census_IsPortableOperatingSystem', 'Census_IsSecureBootEnabled', 'Census_IsTo
uchEnabled', 'Census_IsVirtualDevice', 'Census_MDC2FormFactor', 'Census_MDC2FormFactor_new', 'Census_OSArchitecture', 'Census_O
SBranch', 'Census_OSEdition', 'Census_OSInstallTypeName', 'Census_OSSkuName', 'Census_OSWUAutoUpdateOptionsName', 'Census_Power
PlatformRoleName', 'Census_PrimaryDiskTypeName', 'Census_ProcessorCoreCount', 'Firewall', 'HasTpm', 'IsProtected', 'IsSxsPassiv
eMode', 'OsPlatformSubRelease', 'OsVer', 'Platform', 'Processor', 'SMode', 'SkuEdition', 'SmartScreen', 'Wdft_IsGamer']
  'New categorical_feature is {}'.format(sorted(list(categorical_feature))))
```

```
Out[22]: LGBMClassifier(boosting_type='gbdt', class_weight=None,
                        colsample_bytree=0.6110437067662637, importance_type='split',
                        learning_rate=0.0106, max_depth=-1, min_child_samples=295,
                        min_child_weight=0.001, min_split_gain=0.0, n_estimators=12000,
                        n_jobs=-1, num_leaves=160, objective='binary', random_state=50,
                        reg_alpha=0.6321152748961743, reg_lambda=0.6313659622714517,
                        silent=True, subsample=0.8202307264855064,
                        subsample_for_bin=80000, subsample_freq=0)
```

Fig4.7.4: Training the model

5. SYSTEM DESIGN

CHAPTER 5

SYSTEM DESIGN

Design Overview

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Beginning, once systemrequirement have been specified and analyzed, system design is the first of the three technicalactivities design, code and test that is required to build and verify software.

The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage. The purpose of the design phase is to plan a solution of the problem specified by the requirement document. This phase is the first step in moving from the problem domain to the solution domain. In other words, starting with what is needed; design takes us toward how to satisfy the needs. The design of a system is perhaps the most critical factor affection the quality of the software; it has a major impact on the later phase, particularly testing, maintenance. The output of this phase is the design document. This document is similar to a blueprint for the solution and is used later during implementation, testing and maintenance. The design activity is often divided into two separate phases System Design and Detailed Design.

System Design also called top-level design aims to identify the modules that shouldbe in the system, the specifications of these modules, and how they interact with each other to produce the desired results. At the end of the system design all the major data structures, file formats, output formats, and the major modules in the system and their specifications are decided.

During, Detailed Design, the internal logic of each of the modules specified in system

design is decided. During this phase, the details of the data of a module are usually specified in a high-level
design description language, which is independent of the target language in which thesoftware will eventually be implemented.
In system design the focus is on identifying the modules, whereas during detailed design the focus is on designing the logic for each of the modules. In other works, in system design the attention is on what components are needed, while in detailed design how the components can be implemented in software is the issue.

Design is concerned with identifying software components specifying relationships among components. Specifying software structure and providing blue print for the document phase. Modularity is one of the desirable properties of large systems. It implies that the system is divided into several parts. In such a manner, the interaction between parts is minimal clearly specified.

During the system design activities, Developers bridge the gap between the requirements specification, produced during requirements elicitation and analysis, and the system that is delivered to the user. Design is the place where the quality is fostered in development. Software design is a process through which requirements are translated into a representation of software.

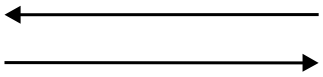
UML Design Overview

Data Flow Diagrams:

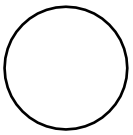
A graphical tool used to describe and analyze the moment of data through a system manual or automated including the process, stores of data, and delays in the system. Data Flow Diagrams are the central tool and the basis from which other components are developed. The transformation of data from input to output, through processes, may be described logically and independently of the physical components associated with the system. The DFD is also know as a data flow graph or a bubble chart.

DFDs are the model of the proposed system. They clearly should show the requirements on which the new system should be built. Later during design activity this is taken as the basis for drawing the system's structure charts. The Basic Notation used to create a DFD's are as follows:

1. Dataflow: Data move in a specific direction from an origin to a destination.

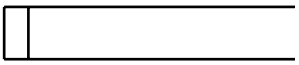


2. Process: People, procedures, or devices that use or produce (Transform) Data. The physical component is not identified



3. Source: External sources or destination of data, which may be People, programs, organizationsor other entities.

4. Data Store: Here data are stored or referenced by a process in the System.



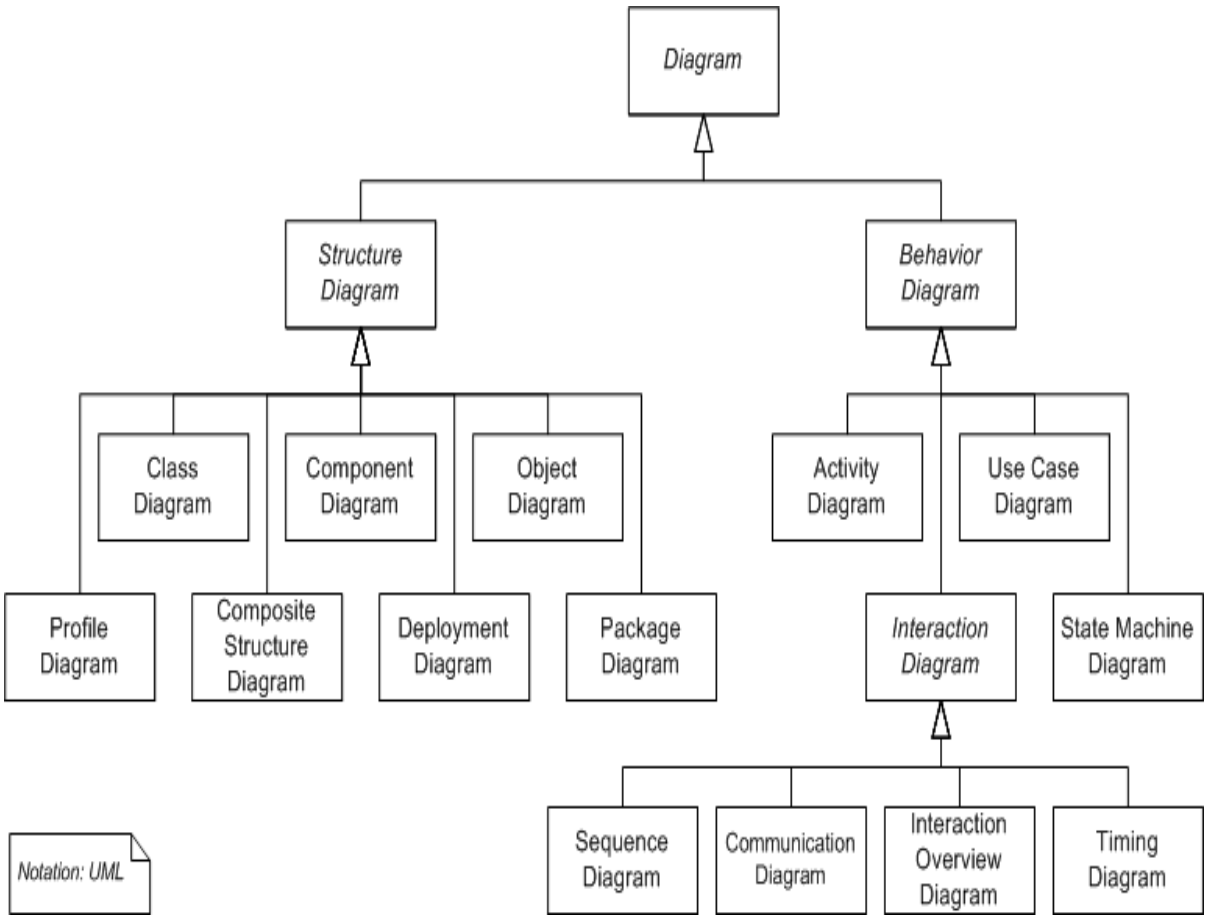


Fig 5.1: UML Diagrams

UML combines best techniques from data modeling (entity relationship diagrams), business modeling (work flows), object modeling, and component modeling. It can be used with all processes, throughout the software development life cycle, and across different implementation technologies. UML has synthesized the notations of the Booch method, the Object-modeling technique (OMT) and Object-oriented software engineering (OOSE) by fusing them into a single, common and widely usable modeling language. UML aims to be a standard modeling language which can model concurrent and distributed systems.

Use Case Diagram

Draw use cases using ovals. Label with ovals with verbs that represent the system's functions.

Actors: Actors are the users of a system. When one system is the actor of another system, label the actor system with the actor stereotype.

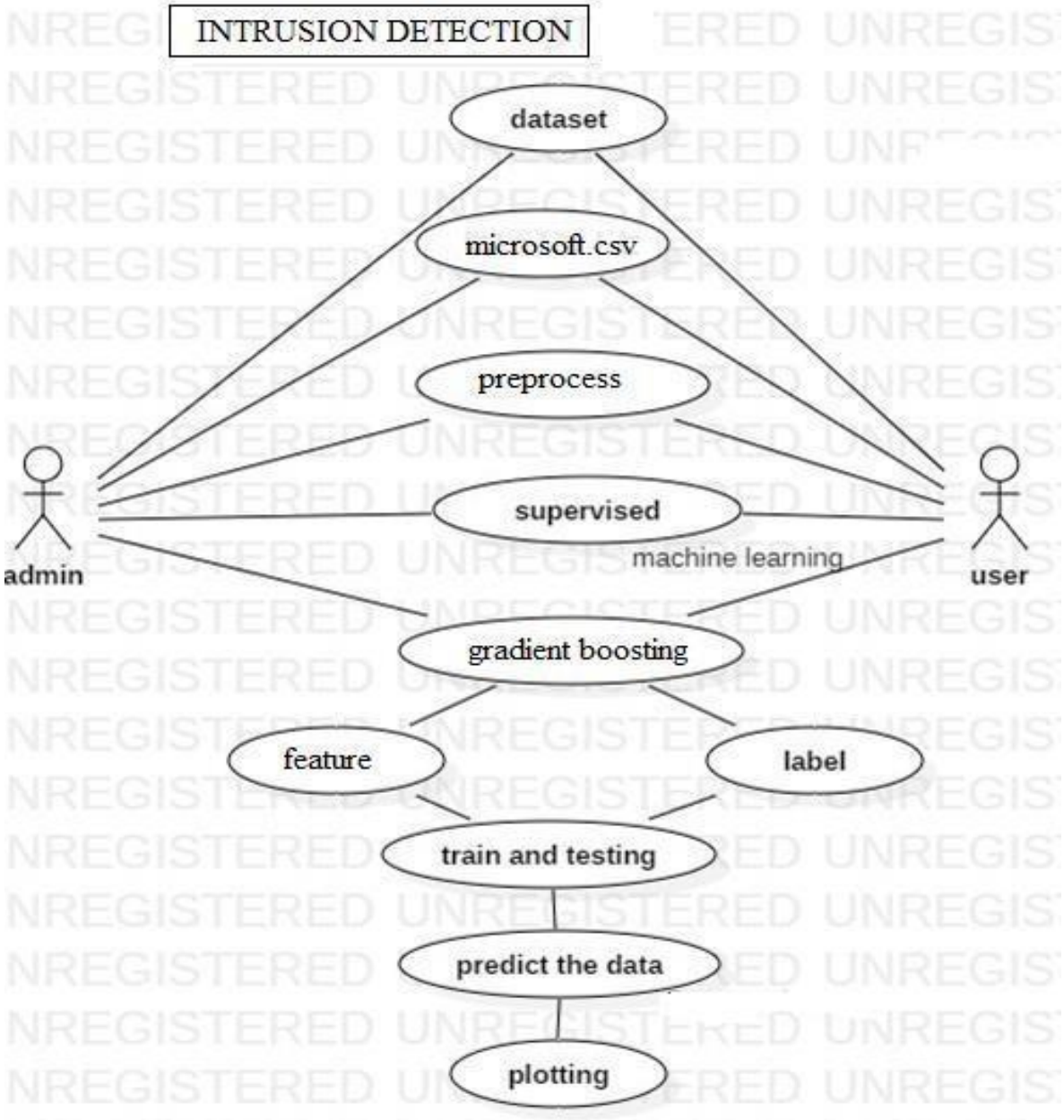


Fig 5.1.1: Use Case Diagram

Class Diagram

Class diagrams are the backbone of almost every object-oriented method including UML. They describe the static structure of a system.

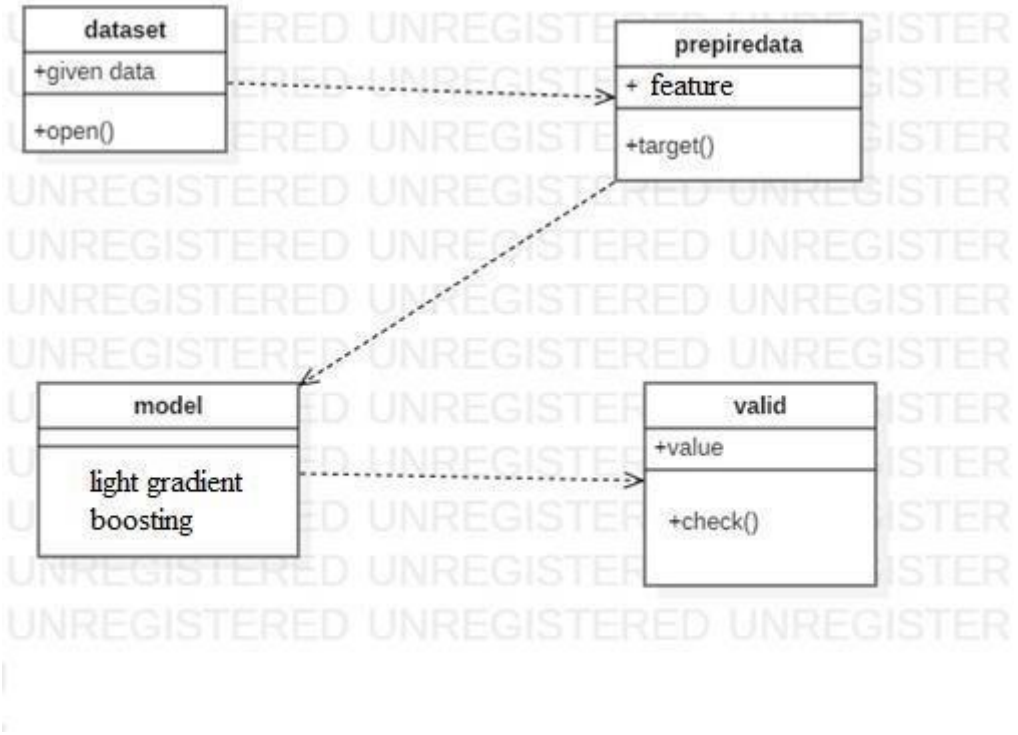
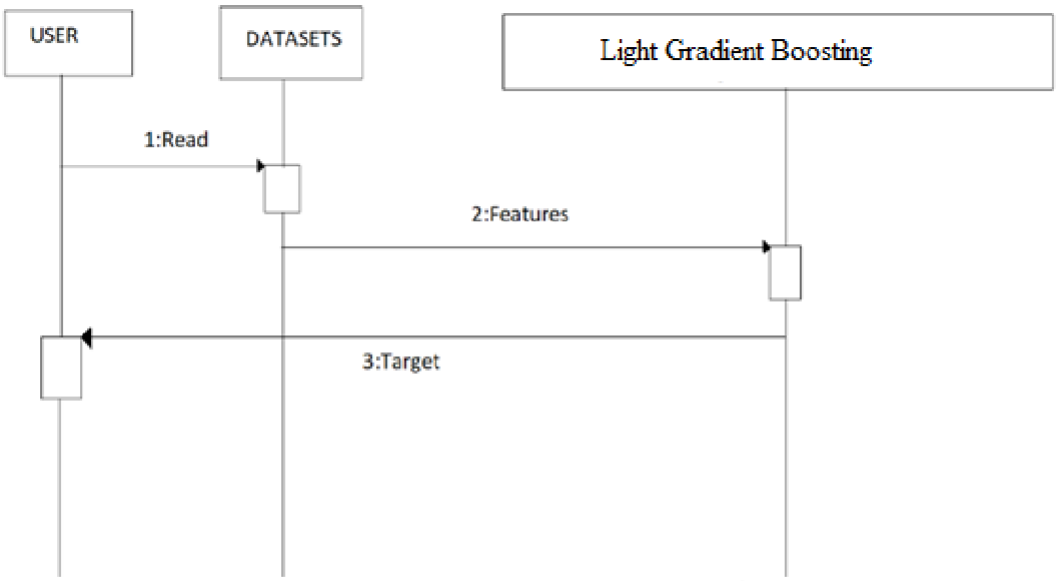


Fig 5.1.2: Class Diagram

Sequence Diagram

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows, as parallel vertical lines ("lifelines"), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchangedbetween them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.



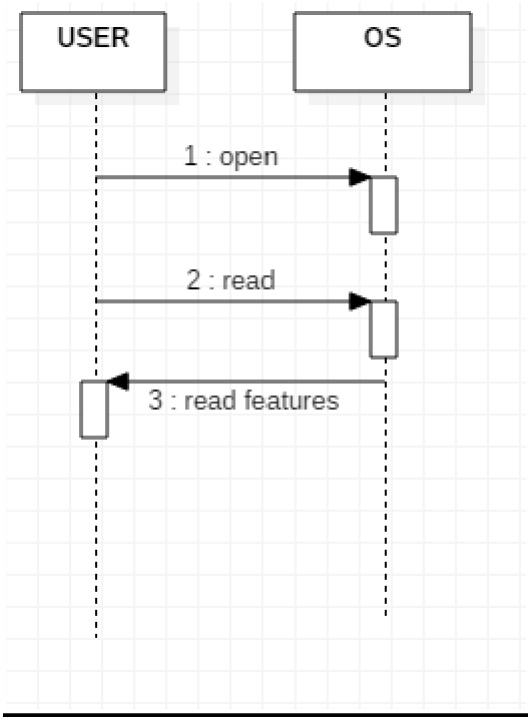


Fig 5.1.3: Sequence Diagram

ACTIVITY DIAGRAM

Activity diagrams are graphical representations of Workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

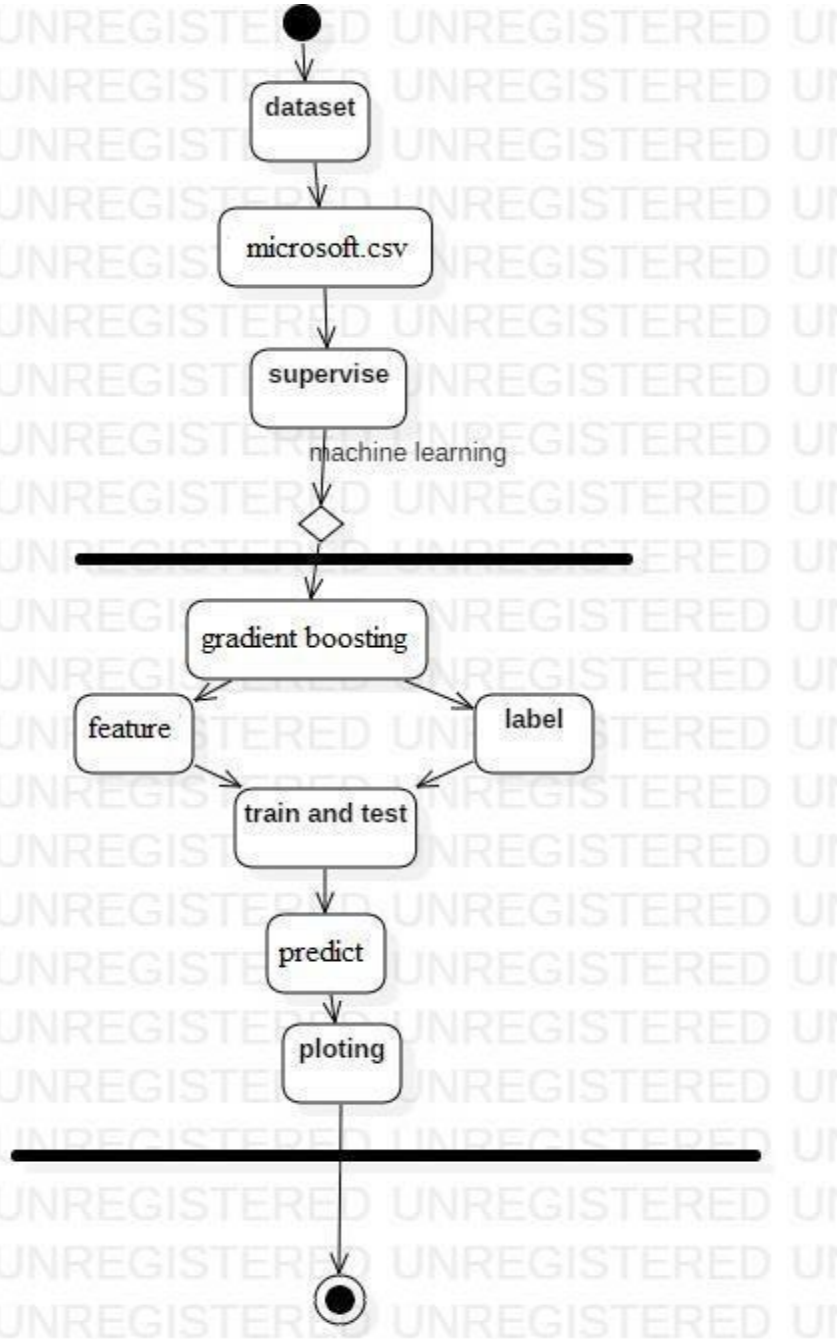


Fig 5.1.4: Activity Diagram

COMPONENT DIAGRAM

Component diagrams are essentially class diagrams that focus on a system's components that often used to model the static implementation view of a system. It does not describe the functionality of the system.

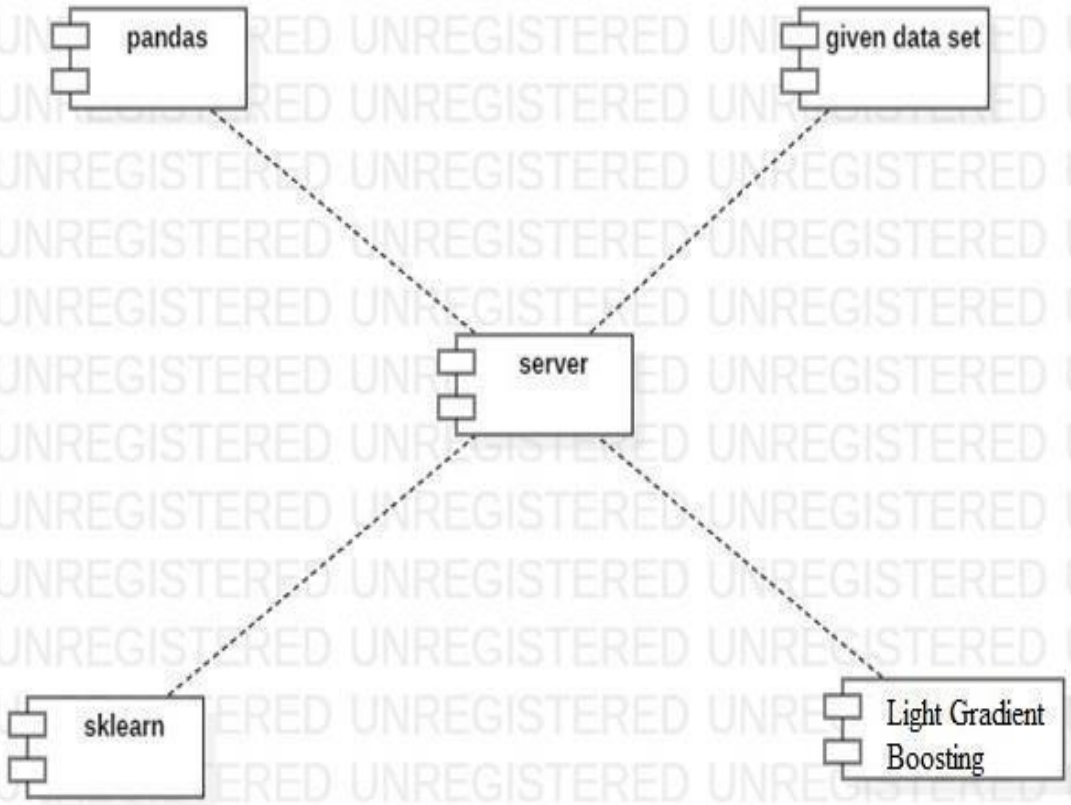


Fig 5.1.5: COMPONENT DIAGRAM

DEPLOYMENT DIAGRAM

Deployment diagrams depict the physical resources in a system including nodes, components, and connections.

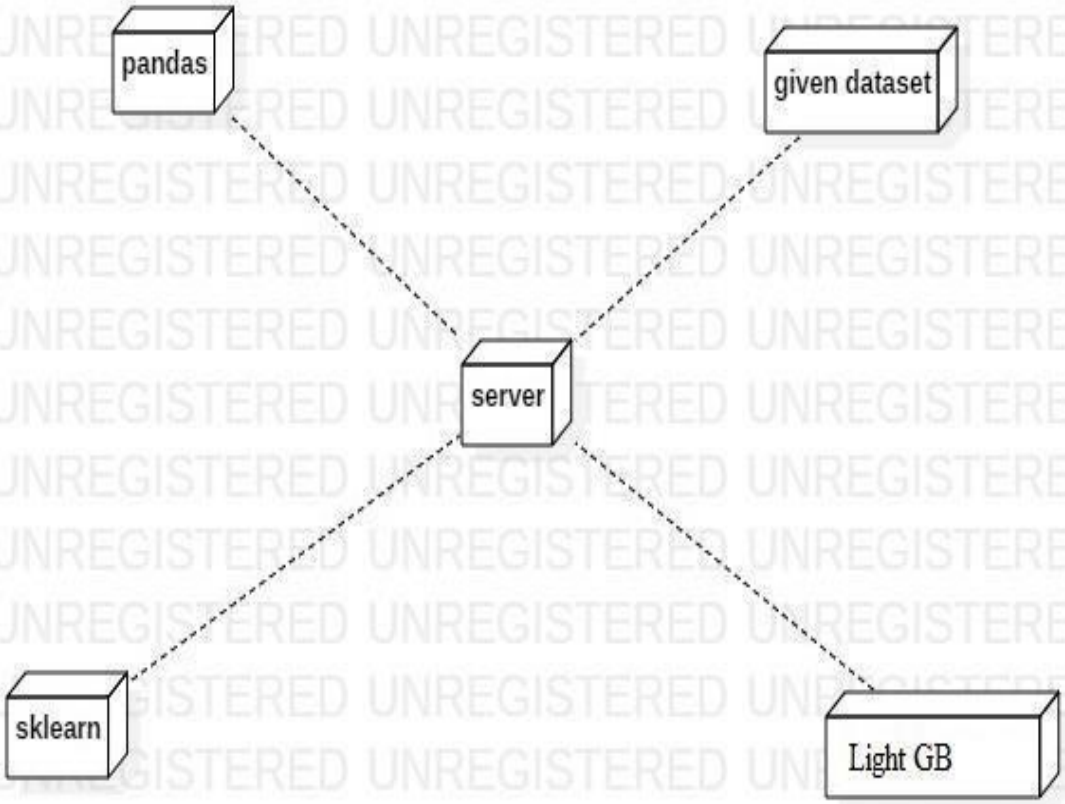


Fig 5.1.6: DEPLOYMENT DIAGRAM

6. IMPLEMENTATION

CHAPTER 6

IMPLEMENTATION

6.1 Outputs Screenshots

```
Classification Report for Light GBM:
              precision    recall  f1-score   support

           0       0.64      0.65      0.65     49659
           1       0.64      0.64      0.64     49341

    accuracy                           0.64     99000
   macro avg       0.64      0.64      0.64     99000
weighted avg       0.64      0.64      0.64     99000

Confusion Matrix of Light GBM:
 [[32112 17547]
 [17764 31577]]
```

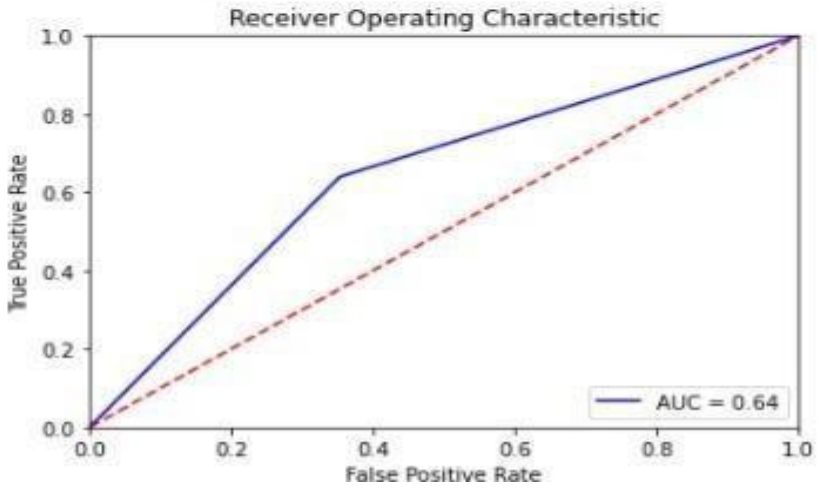
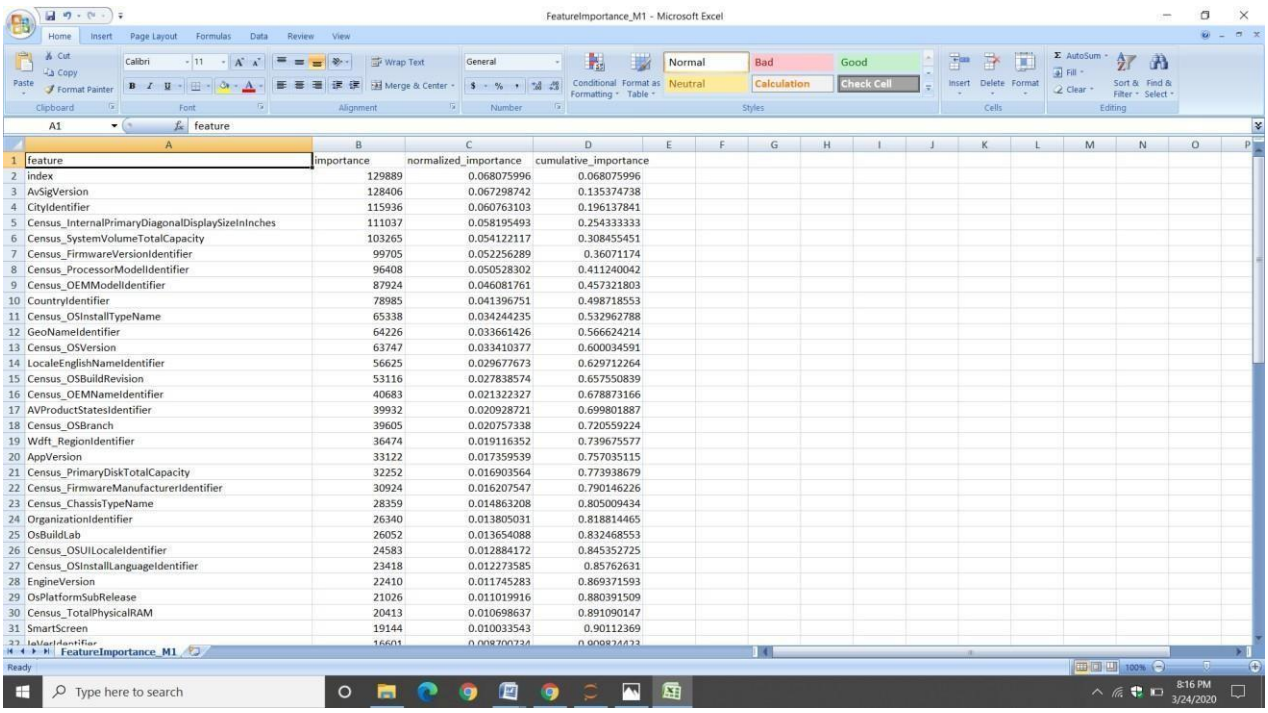


Fig6.1.1: Prediction using ROC Cuve

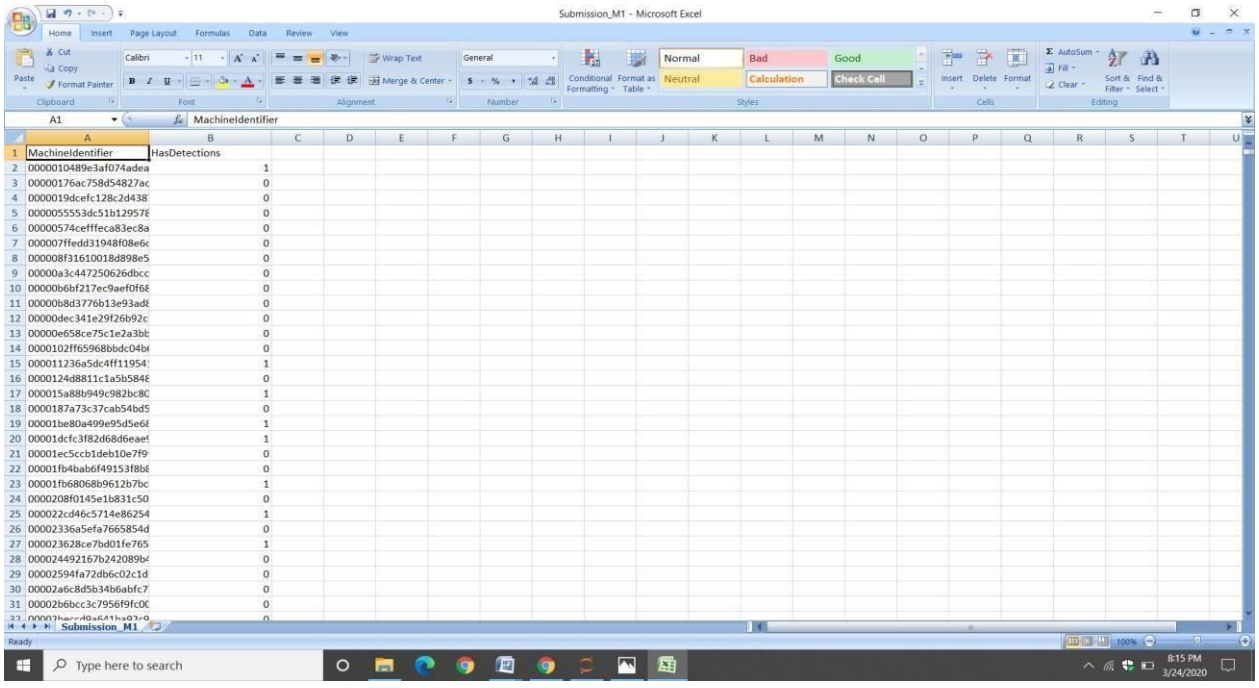
```
In [23]: model.score(X_test,y_test)
Out[23]: 0.6428989898989899
```

Fig6.1.2: Accuracy of the system



feature	importance	normalized_importance	cumulative_importance
index	129889	0.068075996	0.068075996
AvSigVersion	128406	0.067298742	0.135374738
CityIdentifier	115936	0.060763103	0.196137841
Census_InternalPrimaryDiagonalDisplaySizeInInches	111037	0.058195493	0.254333333
Census_SystemVolumeTotalCapacity	103265	0.054122117	0.308455451
Census_FirmwareVersionIdentifier	99705	0.052256289	0.36071174
Census_ProcessorModelIdentifier	96408	0.050528302	0.411240042
Census_OEMModelIdentifier	87924	0.046081761	0.457321803
CountryIdentifier	78985	0.041396751	0.498718553
Census_OSInstallTypeName	65338	0.034244235	0.532962788
GeoNameIdentifier	64226	0.033661426	0.566624214
Census_OSVersion	63747	0.033410377	0.600034591
LocaleEnglishNameIdentifier	56625	0.029677673	0.629712264
Census_OSBuildRevision	53116	0.027838574	0.657550839
Census_OEMNameIdentifier	40683	0.021322327	0.678873166
AVProductStatesIdentifier	39932	0.020928721	0.699801887
Census_OSBranch	39605	0.020757338	0.720559224
Wdft_RegionIdentifier	36474	0.019116352	0.739675577
AppVersion	33122	0.017359539	0.757035115
Census_PrimaryDiskTotalCapacity	32252	0.016903564	0.773938679
Census_FirmwareManufacturerIdentifier	30924	0.016207547	0.790146226
Census_ChassisTypeName	28359	0.014863208	0.805009434
OrganizationIdentifier	26340	0.013805031	0.818814465
OsBuildLab	26052	0.013654088	0.832468553
Census_OSUILocaleIdentifier	24583	0.012884172	0.845352725
Census_OSInstallLanguageIdentifier	23418	0.012273585	0.85762631
EngineVersion	22410	0.011745283	0.869371593
OsPlatformSubRelease	21026	0.011019916	0.880391509
Census_TotalPhysicalRAM	20413	0.010698637	0.891090147
SmartScreen	19144	0.010033543	0.90112369

Fig6.1.3: Importance of Selected Features



MachineIdentifier	HasDetections
0000010489e3af074adea	1
00000176ac758d54827ac	0
0000019dcefc128c2d438	0
0000055553dc51b12957	0
00000574cefffeca83ec8a	0
000007ffedd31948f08e6	0
000008f31610018d898e5	0
00000a3c447250626dbcc	0
00000b6bf217ec9aef0f68	0
00000b8d3776b13e93ad8	0
00000dec341e29f26b92c	0
00000e658ce75c1e2a3bb	0
0000102ff65968bbdc04b	0
000011236a5dc4ff11954	1
0000124d8811c1a5b5848	0
000015a88b949c982bc80	1
0000187a73c37cab54bd5	0
00001be80a499e95d5e68	1
00001dcfc3f82d68d6eae5	1
00001ec5ccb1deb10e7f9	0
00001fb4bab6f49153f8b8	0
00001fb68068b9612b7bc	1
0000208f0145e1b831c50	0
000022cd46c5714e86254	1
00002336a5efa7665854d	0
000023628ce7bd01fe765	1
0000244921672b242089b4	0
00002594fa72db6c02c1d	0
00002a6c8d5b34b6abfc7	0
00002b6bcc3c7956f9fc0	0

Fig6.1.4: Final Output to know whether the system is intruded or not

7. SYSTEM TESTING

CHAPTER 7

SYSTEM

TESTING

Testing

The aim of testing is often to demonstrate that a program works by showing that it has no errors. The basic purpose of testing phase is to detect the errors that may be present in the program. Hence one should not start testing with the intent of showing that a program works but the intent should be to show that a program doesn't work. Testing is the purpose of executing a program with the intent of finding others.

Testing Objectives:

The main objective of testing is to uncover a host of errors, systematically and withminimum effort and time, Stating formally, we can say

- Testing is a process of executing a program for finding an error.
- A successful test is one that uncovers as yet undiscovered errors.
- A good test case is one that has high probability of finding errors.
- The tests are inadequate to detect possibly present errors.
- The software more or less confirms to the quality.

Types of Testing:

- Unit Testing.
- Integration Testing.
- System Testing.
- Acceptance Testing.

Unit Testing:

Unit testing focuses verification effort on the smallest unit of software i.e. module. Using the detailed design and the process specification testing is done to uncover errors within the boundary

of the module. All modules must be successful in the unit test before

the start of the integration testing begins.

In this project each service can be thought of a module. There are so many modules like login, Admin, Faculty, Student. Giving different sets of inputs has tested each module. When developing the module as a well as finishing the development so that each module work without any error.

Integration Testing:

After the unit testing we have to perform integration testing. The goal is to here is to see if modules can be integrated properly, the emphasis being on testing interfaces between modules. This testing activity can be considered as testing the design and hence the emphasis on testing module interactions.

In this project integrating the entire module forms the main system. When integrating all the modules I have checked whether the integration effects working of any of the services by giving different combinations of inputs with which the two services run perfectly before integration.

Analogy:

During the process of manufacturing a ballpoint pen, the cap, the body, the tail and clip, the ink cartridge and the ballpoint are produced separately and unit tested separately. When two or more units are ready, they are assembled and Integration Testing is performed. For example, whether the cap fits into the body or not.

Method:

Any of Black Box Testing, White Box Testing and Gray Box Testing methodscan be used. Normally, the method depends on your definition of ‘unit’.

Tasks

- Integration Test Plan
    - Prepare
    - Review
    - Rework
    - Baseline
- Integration Test Cases/Scripts
    - Prepare
    - Review

- Rework
- Baseline
- Integration Test

Integration Testing is the second level of testing performed after Unit Testing and before System Testing. Developers themselves or independent testers perform Integration Testing.

System Testing:

Here the entire software system is tested. The reference document for this process isthe requirements of the document. And the goal as to see if software needs its requirements.

Acceptance Testing:

- Acceptance test is performed with realistic data of the client to demonstrate that the software is working satisfactory. Testing here is focused on external behaviour of the system: the internal logic of program is not emphasized.
- Test cases should be selected so that the largest number of attributes of an aquiline class is exorcized at once. The testing phase is an important path of software development. It is the whether the objectives are met and the user requirements are satisfied.
- In system testing, integration testing passed components are taken as input. The goal of integration testing is to detect any irregularity between the units that are integrated together. System testing detects defects within both the integrated units and the whole system. The result of system testing is the observed behavior of a component or a system when it is tested.
- **System Testing** is carried out on the whole system in the context of either system requirement specifications or functional requirement specifications or in the context of both. System testing tests the design and behavior of the system and also the expectations of the customer.

8. CONCLUSION

CHAPTER- 8

CONCLUSION

This is worked on issue related to Light GBM machine learning algorithm as it assume strong feature independence between attributes so proposed new algorithm which approximates the interactions between attributes by using conditional probabilities. The performance comparison amongst different classifiers with proposed classifier is made in order to understand their effectiveness in terms of various performance measures. From results, it is clear that every attributes in data set is not of equal importance, as we can ignore some attributes over others which does not involve much in intrusion detection. So this study has applied the feature selection techniques and found better results than before.

9. FUTURE ENHANCEMENT

CHAPTER-9

FUTURE ENHANCEMENT

In the digital age, malware have impacted a large number of computing devices. The term malware come from malicious software which is designed to meet the harmful intent of a malicious attacker. To avoid this we have used Microsoft Dataset which consists of large amount of data. So, we will select some of the important features manually that are related to our work, as all the features doesn't have equal importance. So, In future we will try to implement feature selection using soft computing techniques to identify intrusion in adaptive heterogeneous environment.

References

[1] Gnes Kayack, H., Nur Zincir- Heywood, A., and Heywood, M. I.: Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets. Third Annual Conference on Privacy, Security and Trust, (2005).

[2] Balakrishnan, S., and Kannan, V.K.:Intrusion Detection System Using Feature Selection and Classification Technique. International Journal of Computer Science and Application (IJCSA), vol. 3, issue 4, (2014).

[3] Vinchurkar, D. P., and Reshamwala, A.: A Review of Intrusion Detection System Using NN and Machine Learning Technique.Debar, H, Dacier, M., and Wespi, A, A Revised taxonomy for intrusion detection systems, Annales des Telecommunications Vol. 55, No.7–8, 361–378, 2000.

[4] Sommer, R., and Paxson, V.:Outside the Closed World: On Using Machine Learning For Network Intrusion Detection. IEEE Symposium on Security and Privacy, pp. 305-316, (2010).

[5] J Shun and HA Malki, A neural network-based network intrusion detection system. Proc. Fourth IEEE Int Conf Nat Comput 5, 242–246 (2008).ICNC'08.

[6] MM Kabir and MM Islam, K Murase, Using a Neural Network, a new wrapper feature selection approach. NeuroComputing 73, 3273–3283 (2010). Elsevier.

[7] G Xiantai, J Weidong, Z Dao, In Metropolitan Area Networks, a Multi-Agent Scheme for Identification and Containment. J. Electron. (China) 23(2),259–265 (2006).

[8] F Amiri, MMR Yousefi, C Lucas, A Shakery, N Yazdani, For intrusion detection systems, feature selection is based on shared knowledge. J. Network Comput. Appl 34, 1184–1199 (2011).

[9] H Liu, L Yu, Towards combining grouping and clustering feature collection algorithms. IEEE Trans. Knowl. Data Eng. 17, 491–502 (2005).

[10] Scarfone, K., and Mell, P.:Guide to Intrusion Detection and Prevention Systems (IDPS). National Institute Of Standards and Technology. Special Publication February-2007.