A PROJECT REPORT ON

AI-POWERED CYBERSECURITY: MACHINE LEARNING FOR THREAT IDENTIFICATION IN FINANCIAL INSTITUTIONS

Submitted in partial fulfillment of requirements for the award of the degree of

MASTER OF COMPUTER APPLICATIONS

Submitted by:

SINGA SANJEEVARAJU (22091F0047)

Under the Guidance of

Mrs.S TAHSEEN BANU M.Tech

Assistant Professor, Dept. of CSE

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

RAJEEV GANDHI MEMORIAL COLLEGE OF ENGINEERING & TECHNOLOGY
(AUTONOMOUS)
Approved by AICTE, New Delhi; Affiliated to JNTUA-Ananthapuramu,
Accredited by NBA (6-Times); Accredited by NAAC with 'A+' Grade (Cycle-3),New Delhi;
World Bank Funded Institution;Nandyal (Dist)-518501, A.P
(Estd-1995)

YEAR: 2023-2024

Rajeev Gandhi Memorial College of Engineering &Technology

(AUTONOMOUS)

Approved by AICTE, New Delhi; Affiliated to JNTUA-Ananthapuramu,
Accredited by NBA (6-Times); Accredited by NAAC with 'A+' Grade (Cycle-3), New Delhi;
World Bank Funded Institution; Nandyal (Dist)-518501, A.P

(ESTD – 1995)

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

CERTIFICATE

This is to certify that **SINGA SANJEEVARAJU**(22091F0047), of MCA IV-semester, has carried out the major project work entitled "**AI-POWERED CYBER SECURITY MACHINE LEARNING FOR THREAT IDENTIFICATION IN FINANCIAL INSTITUTIONS** " under the supervision and guidance of **Mrs.S TAHSEEN BANU ,** Assistant Professor, CSE Department, in partial fulfilment of the requirements for the award of Degree of **Master of Computer Applications** from **Rajeev Gandhi Memorial College of Engineering & Technology (Autonomous)**, Nandyal is a bonafied record of the work done by him during2023-2024.

Project Guide

Mrs.S TAHSEEN BANU M.Tech
Assistant Professor, Dept. of CSE

Head of the Department

Dr. K. SUBBA REDDY, M.Tech, Ph.D.
Professor& HOD, Dept. of CSE.

Place: Nandyal
Date:

External Examiner

Candidate's Declaration

I hereby declare that the work done in this project entitled **"AI-POWERED CYBERSECURITY: MACHINE LEARNING FOR THREAT IDENTIFICATION IN FINANCIAL INSTITUTIONS"** submitted towards completion of major project in MCA IV-semester at the **Rajeev Gandhi Memorial College of Engineering & Technology**, Nandyal. It is an authentic record of my original work done under the esteemed guidance of **Mrs.S TAHSEEN BANU** Assistant Professor, Department of **Computer Science and Engineering**, RGMCET, Nandyal.

I have not submitted the matter embodied in this report for the award of any other Degree in any other institutions for the academic year 2023-2024**.**

By

(SINGA SANJEEVRAJU)

Dept. of MCA,

RGMCET.

Place: Nandyal
Date:

ACKNOWLEDGEMENT

I manifest our heartier thankfulness pertaining to your contentment over my project guide **Mrs.S TAHSEEN BANU,** Assistant Professor of Computer science Engineering department, with whose adroit concomitance the excellence has been exemplified in bringing out this project to work with artistry.

I express our gratitude to **Dr. K. Subba Reddy garu,** Head of the Department of Computer Science Engineering & Master of Computer Applications, all the **Teaching Staff Members** of the CSE & MCA departments of Rajeev Gandhi Memorial College of Engineering and Technology for providing continuous encouragement and cooperation at various steps of my project successful.

Involuntarily, I am perspicuous to divulge our sincere gratefulness to my Principal, **Dr. T. Jaya Chandra Prasad garu,** who has been observed posing valiance in abundance towards
my individuality to acknowledge my project work tangentially.

At the outset I thank my Honorable **Chairman Dr. M. Santhi Ramudu garu,** for providing us with exceptional faculty and moral support throughout the course.

Finally, I extend our sincere thanks to all the non- teaching **Staff Members** of CSE & MCA Department who have co-operated and encouraged me in making to my project successful.

Whatever one does, whatever one achieves, the first credit goes to the **Parents** be it not for their love and affection, nothing would have been responsible. I see in every good that happens to me their love and blessings.

BY

SINGA SANJEEVRAJU (22091F0047)

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

The increasing interconnectedness of digital assets is leading to an unparalleled surge in cyber attacks. Investments in artificial intelligence-based solutions are necessary if financial institutions are to recognize these dangers and safeguard their assets. When examining intricate financial security threats that are dynamic and often unpredictable, machine learning is a potent tool. Through the utilization of artificial intelligence (AI) technology, such as automated reasoning systems, natural language processing, and algorithms, banks can enhance their comprehension of potential hazards and establish more effective data controls.

This study proposes a machine learning approach to identify cyber security concerns in financial institutions using artificial intelligence.Machine learning algorithms are continuously being enhanced to find data anomalies that could point to a security risk. Financial institutions can use custom-made models that offer actionable insights into both internal and external risks to detect and protect against harmful assaults.

CHAPTER-1

1.INTRODUCTION

1.1 Introduction

1.1 General Introduction:

Financial fraud refers to the use of fraudulent and illegal methods or deceptive tactics to gain financial benefits. Fraud can be committed in different areas of finance, including banking, insurance, taxation, and corporates, and more. Fiscal fraud and evasion, including credit card fraud, tax evasion, financial statement fraud, money laundry, and other types of financial fraud, has become a growing problem. Despite efforts to eliminate financial fraud, its occurrence adversely affects business and society as hundreds of millions of dollars are lost to fraud each year. This significant financial loss has dramatically affected individuals, merchants, and banks.

Nowadays, fraud attempts have increased drastically, which makes fraud detection more important than ever. The Association of Certified Fraud Examiners (ACFE) has announced that 10% of incidents concerning white-collar crime involves falsification of financial statements. They classified occupational fraud into three types: asset misappropriation, corruption, and financial statement fraud. Financial statement fraud resulted in the most significant losses among them.

Although the occurrence frequency of asset misappropriation and corruption is much higher than financial statement fraud, the financial implications of these latter crimes are still far less severe. In particular, as reported in a survey from Eisner Amper, which is among the prominent accounting firms in the U.S., ‘‘the average median loss of financial statement fraud ($800,000 in 2018) accounts for over three times the monetary loss of corruption ($250,000) and seven times as much as asset misappropriation ($114,000)’’.

The focus of this study is on financial statement fraud. Financial statements are documents that describe details about a company, specifically their business activities and financial performance, including income, expenses, profits, loans, presumable concerns that may

emerge later, and managerial comments on the business performance. All firms are obligated to announce their financial statements in a quarterly and annual manner. Financial statements can be used to indicate the performance of a company. Investors, market analysts, and creditors exploit financial reports to investigate and assess the financial health and earnings potentials of a business.

Financial statements consist off our sections; income statement, balance sheet, cash flow statement, and explanatory notes. The income statement places a great emphasis on a company's expenses and revenues during a specific period.The company's profit or net income is provided in this section, which subtracts expenses from revenues.

The balance sheet provides a timely snapshot of liabilities, assets, and stockholders' equity. The cash flow statement measures the extent to which a company is successful in making cash to fund its operating expenses, fund investments, and pay its debt obligations. Explanatory notes are supplemental data that provide clarification and further information about particular items published financial statements of a company.

These notes cover areas including disclosure of subsequent events, asset depreciation, and significant accounting policies, which are necessary disclosures that demonstrate the amounts reported on the financial statements. Financial statement fraud involves falsifying financial statements to pretend the company more profitable than it is, increase the stock prices, avoid payment of the taxes, or get a bank loan.

Fraud triangle in auditing is a framework to demonstrate the motivation behind an individual's decision to commit fraud. The fraud triangle has three elements that increase the risk of fraud: incentive, rationalization, and opportunity, which, together, lead to fraudulent behavior. Auditing professionals have extensively used this theory to explain the motivation behind an individual's decision to commit fraud.

It is indispensable to understand the fraud triangle to evaluate financial fraud. Gupta and Singh suggested that when there are incentives such as the obligation to achieve an outcome or cover losses, the potential for fraud increases. The company will encounter temptations or pressures to adopt fraudulent practices.

Moreover, the lack of inspections or unsuccessful controls provides a favourable occasion for committing fraud. Rationalization happens when the fraudster aims to justify the fraudulent action, and it could be affected by the others and the conditio

1.2 Problem Statement:

Fraud detection refers to the problem of finding patterns in data that do not conform to expected behavior. These nonconforming patterns are often referred to as fraud, outliers, discordant observations, exceptions, aberrations, surprises, peculiarities, or contaminants in different application domains.

1.3 Objectives:

The main objective of our project is,

- To predict or to classify the fraud and non-fraud data from financial statements.
- To implement the machine learning algorithm.
- To enhance the performance analysis.

2.LITERATURE REVIEW

2.1 EXISTING SYSTEM:

Fraudulent financial statements (FFS) are the results of manipulating financial elements by overvaluing incomes, assets, sales, and profits while underrating expenses, debts, or losses. To identify such fraudulent statements, traditional methods, including manual auditing and inspections, are costly, imprecise, and time-consuming. Intelligent methods can significantly help auditors in analyzing a large number of financial statements. In this study, we systematically review and synthesize the existing literature on intelligent fraud detection in corporate financial statements.

In particular, the focus of this review is on exploring machine learning and data mining methods, as well as the various datasets that are studied for detecting financial fraud. We adopted the Kitchen ham methodology as a well-defined protocol to extract, synthesize, and report the results. Accordingly, 47 articles were selected, synthesized, and analyzed. We present the key issues, gaps, and limitations in the area of fraud detection in financial statements and suggest areas for future research.

Since supervised algorithms were employed more than unsupervised approaches like clustering, the future research should focus on unsupervised, semi-supervised, as well as bio-inspired and evolutionary heuristic methods for fraud (fraud) detection.

In terms of datasets, it is envisaged that future research making use of textual and audio data. While imposing new challenges, this unstructured data deserves further study as it can show interesting results for intelligent fraud detection.

2.1.1 DISADVANTAGES:

- The results is low when compared with proposed.
- Time consumption is high.
- Theoretical limits.

2.2 PROPOSED SYSTEM:

In our proposed system, we detect the fraud in financial statements by using the machine learning algorithm. First, we select and view the imported dataset for future purpose. And we get missing values and fill the default values to the dataset. We encoding the label in the dataset. And we split the dataset to the Train and Test data for predict the fraud or non-fraud. Then we use three algorithms for more accuracy, prediction and which is more accurate value.

There are Random forest algorithm, KNN classifiers and Ada-Boost Algorithm. Now, we fit the training data from the dataset. Then we predict the test dataset using training dataset. Then the test values get the results of actual and predicted. And we get the performance of the dataset. It is essential to train the models on data which includes fraud and relevant non fraud.

By using the ML algorithm the system is, to classify the fraud and non-fraud and results shows that the accuracy, precision, recall and f1-score and also prediction. This shows that method used in this project can predict the possibility of fraud accurately in most of the cases. This module is the simple and effective way to avoid such frauds and save those expenditures.

2.2.1 ADVANTAGES

- It is efficient for large number of datasets.
- The experimental result is high when compared with existing system.
- Time consumption is low.
- Provide accurate prediction results.

2.3 LITERATURE SURVEY:

2.3.1 Evaluation of financial statements fraud detection research: A multidisciplinary analysis, 2019.

Author*:* A. Albizia, D. Appelbaum, and N. Rizzotto

Methodology:

Prior research in the fields of accounting and information systems has shed some light on the significant effects of financial reporting fraud on multiple levels of the economy.

In this paper, we compile prior multi-disciplinary literature on financial fraud detection. Financial reporting fraud detection efforts and research may be more impactful when the findings of these different domains are combined. We anticipate that this research will be valuable for academics, analysts, regulators, practitioners, and investors.

Advantages:

- Reduced Manual power.
- Low cost.

Disadvantages:

- Too Many False Negatives.
- Run to failure prediction is low.

2.3.2 Interpretable fuzzy rule-based systems for detecting financial statement fraud, 2019

Author*:* P. Hajek

Methodology:

Systems for detecting financial statement frauds have attracted considerable interest in computational intelligence research. Diverse classification methods have been employed to perform automatic detection of fraudulent companies. However, previous research has aimed to develop highly accurate detection systems, while neglecting the interpretability of those systems.

Here we propose a novel fuzzy rule-based detection system that integrates a feature selection component and rule extraction to achieve a highly interpretable system in terms of rule complexity and granularity. Specifically, we use a genetic feature selection to remove irrelevant attributes and then we perform a comparative analysis of state-of-the-art fuzzy rule-based systems, including FURIA and evolutionary fuzzy rule-based systems.

Here, we show that using such systems leads not only to competitive accuracy but also to desirable interpretability. This finding has important implications for auditors and other users of the detection systems of financial statement fraud.

Advantages:

- Avoid the over fitting from the dataset.

Disadvantages:

- It can be intimidating.

2.3.3 An application of ensemble random forest classifier for detecting financial statement manipulation of Indian listed companies, 2019

Author*:* H. Patel, S. Parikh, A. Patel, and A. Parikh

Methodology:

A rising incidents of financial frauds in recent time has increased the risk of investor and other stakeholders. Hiding of financial losses through fraud or manipulation in reporting and hence resulted into erosion of considerable wealth of their stakeholders. In fact, a number of global companies like WorldCom, Xerox, Enron and number Indian companies such as Satyam, Kingfisher and Deccan Chronicle had committed fraud in financial statement by manipulation. Hence, it is imperative to create an efficient and effective framework for detection of financial fraud.

This can be helpful to regulators, investors, governments and auditors as preventive steps in avoiding any possible financial fraud cases. In this context, increasing number of research these days have started focusing on developing systems, models and practices to detect fraud in early stage to avoid the any attrition of investor's wealth and to reduces the risk of financing.

Current study, the researcher has attempted to explore the various 42 modeling techniques to detect fraud in financial statements (FFS). To perform the experiment, researcher has chosen 86 FFS and 92 non-fraudulent financial statements (non FFS) of manufacturing firms. The data were taken from Bombay Stock Exchange for the dimension of 2008-2011. Auditor's report is considered for classification of FFS and Non-FFS companies.

T-test was applied on 31 important financial ratios and 10 significant variables were taken in to consideration for data mining techniques. 86 FFS and 92 non-FFS during 2008-2017 were taken for testing data set. Researcher has trained the model using data sets.

Then, the trained model was applied to the testing data set for the accuracy check. Random forest gives best accuracy. Here, modified random forest model was developed with improved accuracy.

Advantages:

- Change of detecting unknown prediction.
- Fraud Detection more efficient than fraud detection, if fraud detection file is large.

Disadvantages:

- Run to failure prediction is low.
- Reliability is unclear.

2.3.4 Detecting fraudulent financial statements for the sustainable development of the socio-economy in China: A multi-analytic approach, 2019

Author*:* J. Yao, Y. Pan, S. Yang, Y. Chen, and Y. Li

Methodology:

Identifying financial statement fraud activities is very important for the sustainable development of a socio-economy, especially in China's emerging capital market. Although many scholars have paid attention to fraud detection in recent years, they have rarely focused on both financial and non-financial predictors by using a multi-analytic approach.

The present study detected financial statement fraud activities based on 17 financial and 7 non-financial variables by using six data mining techniques including support vector machine (SVM), classification and regression tree (CART), back propagation neural network (BP-NN), logistic regression (LR), Bayes classifier (Bayes) and K-nearest neighbor (KNN).

Specifically, the research period was from 2008 to 2017 and the sample is companies listed on the Shanghai stock exchange and Shenzhen stock exchange, with a total of 536 companies of which 134 companies were allegedly involved in fraud. The stepwise regression and principal component analysis (PCA) were also adopted for reducing variable dimension. The experimental results show that the SVM data mining technique has the highest accuracy across all conditions, and after using stepwise regression, 13 significant variables were screened and the classification accuracy of almost all data mining techniques was improved.

However, the first 16 principal components transformed by PCA did not yield better classification results. Therefore, the combination of SVM and the stepwise regression dimensionality reduction method was found to be a good model for detecting fraudulent financial statements. Identifying financial statement fraud activities is very important for the sustainable development of a socio-economy, especially in China's emerging capital market.

Although many scholars have paid attention to fraud detection in recent years, they have rarely focused on both financial and non-financial predictors by using a multi-analytic

approach. present study detected financial statement fraud activities based on 17 financial and 7 non-financial variables by using six data mining techniques including support vector machine (SVM), classification and regression tree (CART), back propagation neural network (BP-NN), logistic regression (LR), Bayes classifier (Bayes) and K-nearest neighbor (KNN).

The experimental results show that the was found to be a good model for detecting fraudulent financial statements. Fraud detection in a dynamic data stream is a challenging task. The endless bound and high arriving rate of data prohibits fraud detection models to store all observations in memory for processing. In addition, the dynamically moving properties of the data stream exhibit concept drift.

While recent studies focus on feature extraction for fraud detection, majority of them assume data stream are static ignoring the possibility of concept drift occurring. Fraud detection models must operate efficiently in order to deal with high volume and velocity data, that is to have low complexity and to learn incrementally from each arriving observation.

Incremental learning allows the model to adapt to concept drift. In cases where drifting rate is higher than adaptation rate, the capability to detect concept drift and retraining a new model is much preferable to minimize the performance losses. In this paper, we propose MIR MAD, an approach based on multiple incremental robust Mahala Nobis estimators that is efficient, learns incrementally and has the capability to detect concept drift.

Mir Mad is fast, can be initialized with small amount of data, and is able to estimate the drift location on the data stream. Our empirical results show that MIR MAD achieves state-of-the-art performance and is significantly faster. We also performed a case study to show that detecting concept drift is critical to minimize the reductio Fraud detection in a dynamic data stream is a challenging task.

The endless bound and high arriving rate of data prohibits fraud detection models to store all observations in memory for processing. In addition, the dynamically moving properties of the data stream exhibit concept drift. While recent studies focus on feature

extraction for fraud detection, majority of them assume data stream are static ignoring the possibility of concept drift occurring.Fraud detection models must operate efficiently in order to deal with high volume and velocity data, that is to have low complexity and to learn incrementally from each arriving observation. Incremental learning allows the model to adapt to concept drift. In cases where drifting rate is higher than adaptation rate, the capability to detect concept drift and retraining a new model is much preferable to minimize the performance losses.

Advantages:

- Rate of missing report is low.
- Simple and Effective method.

Disadvantages:

- Needs to be trained, and trained model carefully otherwise tends to be false positive
- Low Accuracy rate.

3.SYSTEM DESIGN

3.1 Model of Procedure Employed with Justification

SDLC:

3.2 Software Development Life Cycle Analysis

The Spiral Model is an iterative approach to software development that blends aspects of waterfall and prototyping approaches The Spiral Model could be used in this: project in the following ways**.**

Identification of Goals and Requirements (Planning):

Specify the goals, limitations, and specifications of the project for the AI-powered e-vaccination system.

Risk Analysis and Evaluation (Risk Assessment):

Determine possible dangers connected to user acceptability, data security, and AI integration.

Analyze the consequences and probability of these hazards.

Engineering Prototype Development:

Create a simple e-vaccination system prototype to highlight important aspects.This possibly a streamlined form of the system with an emphasis on key features like AI Chabot interaction and appointment scheduling.

Build and Test (Engineering and Testing):

Create the essential e-vaccination system components, such as the database configuration and AI integration. Test everything thoroughly to find and address any problems or bugs.

Risk Analysis and Evaluation (Risk Assessment - Iteration 1):

Reassess the hazards that have been recognized, taking into account comments and insights from the testing and prototyping stages.

Risk Analysis and Evaluation (Risk Assessment - Iteration 2):

Reevaluate risks in light of developments and new information.

Complete System Development (Engineering - Iteration 2):

Create the user interfaces, data security protocols, and all anticipated features of the whole e-vaccination system.

Comprehensive Testing (Testing - Iteration 2):

the completely developed system thoroughly to make sure it satisfies all requirements and is error-free.

The fully completed e-vaccination system should be implemented in a controlled setting, like a pilot program for a particular community.

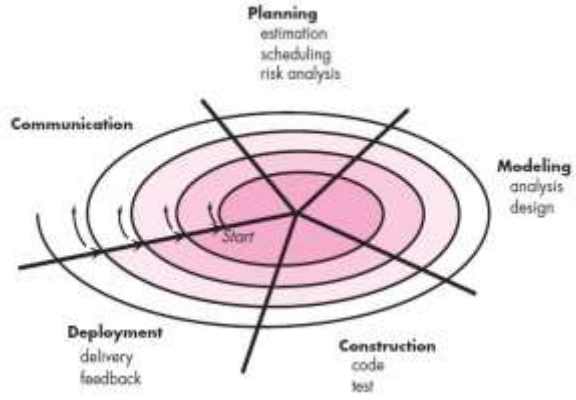


Fig.1:Spiral Model

Evaluation (Review and Evaluation):

Get input from stakeholders and users. Evaluate the security, usability, and performance of the system.

Maintenance and Enhancement (Maintenance and Support):

Give the system continuous maintenance and assistance. Make ongoing changes and improvements by utilizing feedback and assessments.

Risk Assessment - Iteration 3:

Risk Analysis and Evaluation- As long as the system is being actively used, keep evaluating and mitigating risks.

Throughout the course of a project, the Spiral Model facilitates iterative development by integrating risk management and feedback.

3.3 Feasibility Study:

In this stage, The project's viability is evaluated and a business proposal with a basic project plan and some cost estimates is presented. During system analysis, The recommended system's feasibility must be examined. This is to make certain that the suggested approach won't put endues train on the company. A feasibility study requires a basic understanding of the main requirements of the system. The feasibility analysis takes into consideration three primary factors: social, technological, and economic feasibility.

3.3.1 Economical Feasibility:

This study aims to assess the possible financial impact of the system on the organization. How much money the corporation can devote to system research and development is limited. The costs have to make sense. Given the financial limits, the intended system could also be implemented because the majority of the technologies were publicly available. Just the items that can be customized need to be bought.

3.3.2 Technical Feasibility:

The aim of this research is to assess the system's technological requirements, or technical feasibility. Any system's development shouldn't put an undue burden on the available technical resources. The availability of technological resources will consequently be in high demand. As a result, the client will need to follow strict guidelines. Because implementing the intended system will only require minimal or nonexistent adjustments, it must have the fewest requirements possible.

3.3.3 Social Feasibility:

The study's assessment is one of its objectives. The system's level of user acceptability. This entails instructing the user on the proper usage of the technology. The user needs to accept the system as a necessity rather than a threat. The methods employed to acquaint and instruct users about the system will dictate the degree of acceptance among them. Being the final user of the system, he needs more confidence to give some incisive feedback, which is greatly appreciated.

The SDLC is centered around the Software Development Life Cycle. In order to produce high-quality software, the software industry adopts it as a standard. Stages in the Life Cycle of Software Development: Testing, Design, Coding, Analysis, and Maintenance.

3.4 Project Requirements

3.4.1 Functional requirements:

- User interface with a graph.

3.4.2 Non-Functional prerequisites

Maintainability:

- It helps satisfy new needs and ease maintenance in the future. Expansion of our project is possible.

Robustness:

- The ability to tolerate strain, stress, or adjustments to protocol or situation is known as robustness. It is also provided by our project.

Dependability:

- The capacity of an individual or system to execute and sustain its operations under various conditions is known as reliability. It is also furnished by our project.

Size:

- An application's size is important; a smaller size corresponds with higher efficiency. We have created a database with a 5.05 MB size.

Speed:

- A high speed is advantageous as it means less lines of code. Power Consumption: In systems that run on batteries, power is crucial.
- The customer cannot specify the permitted wattage during the requirement stage, although power can be specified in terms of battery life.
- Our code is less in lines, which means the CPU will run it faster and use less power.

3.4.3 MODULES:

- Data selection
- Data preprocessing
- Data splitting
- Classification
- Prediction
- Performance Metrics
- Graph Comparison

3.4.4 Modules Description:

3.4.5 Data Selection:

The dataset was where the input data was gathered. repository like UCI Repository. ¬ In this process, the input data have some columns like step, type, amount, nameOrig, balanceOrig, name Dest, balance Dest, is Flagged Fraud, etc. In our collected dataset was analyzed in this procedure employing pandas.

3.4.6 Preprocessing:

Data pre-processing is the removal of unwanted information from a dataset. Utilizing pre-processing data transformation techniques, the dataset is converted into a structure appropriate for machine learning. This step also includes cleaning the dataset by removing any extraneous or corrupted data in order to increase its accuracy and efficiency.

Erroneous data removal Removing data that is missing: Null values, which include absent During this operation, 0 is used to replace values and Nan values. To remove all abnormalities, duplicate values, and missing values, the data was cleansed. Label Encoding: During this process, the string values are converted into integers so that additional predictions may be made.

3.4.7 Data Splitting

Machine learning requires data in order for learning to take place. In this instance, the training and testing datasets are kept apart, yet test data are still required to evaluate the algorithm's performance.Training and testing must be separated in our procedure.

Dividing accessible data into two halves, typically for cross-validation objectives, is known as data splitting. Part of the data is utilized for the development of a predictive model, while the remaining portion is used to assess the model's efficacy.

Training and testing must be separated in our procedure. Dividing accessible data into two halves, typically for cross-validation objectives, is known as data splitting. Part of the data is utilized for the development of a predictive model, while the remaining portion is used to assess the model's efficacy.

3.4.8 Classifications

Random Forest Algorithm:

Random forest is a machine learning method for fraud detection. This method separates outliers from the rest of the data using unsupervised learning to identify fraud.

Random Forest isolates the fraudulent data points from the non-fraud data points instead of profiling them. The trees in the isolation forest are usually substantially shorter for fraud data points than for normal data points.

KNN Algorithm:

The supervised learning method is the foundation of K-Nearest Neighbor, one of the most fundamental machine learning algorithms. Assuming that the new case and data are equivalent to the existing cases, the K-NN method places the new into he group that most closely corresponds with the existing categories. After preserving all relevant information, a new data point is classified using the K-NN algorithm based on similarity.

This suggests that newly discovered data can be swiftly classified into an appropriate category by the K-NN algorithm. Regression can also be done using the K-NN approach, albeit classification issues are its primary application. The K-NN algorithm is non-parametric, which means it doesn't make any presumptions about the data that it uses.

The non-parametric nature of K-NN prevents it from making any assumptions about the underlying data. Rather than learning quickly from the training batch, this technique is frequently called a lazy learner because it retains the dataset and uses it for data classification in the training phase, the CNN algorithm keeps track of the data and groups together recently obtained data that are very similar to one another.

Ada Boost Algorithm:

AdaBoost is an ensemble approach for machine learning. It is sometimes called flexible boosting. The most popular algorithm using AdaBoost decision trees are classifications of decision trees that have only one split, or level. Decision Stumps is another term for these trees. Improving, decision tree performance in binary classification tasks is the best use of AdaBoost.

The method's original creators, Freund and Schapiro, referred to AdaBoost as AdaBoost.M1. Since it is employed AdaBoost decision trees are classifications of decision trees that have only one split, or level. Decision Stumps is another term for these trees. Ada Boost decision trees are classifications of decision trees that have only one split, or level. Decision Stumps is another term for these trees.

Prediction:

- Predict the dataset values are Non Fraud/Fraud by using classification algorithm.

3.4.9 Performance Metrics:

The Final Result will get generated based on the overall classification and prediction. The performance of this proposed approach is evaluated using some measures like,

Accuracy:

- Accuracy of classifier refers to the ability of classifier. It predicts the class label correctly and the accuracy of the predictor refers to how well a given predictor can guess the value of predicted attribute for a new data.

  AC= (TP+TN)/ (TP+TN+FP+FN)

Precision:

- Precision is defined as the number of true positives divided by the number of true positives plus the number of false positives.
- Precision=TP/ (TP+FP).

Recall:

- Recall is the number of correct results divided by the number of results that should have been returned. In binary classification, recall is called sensitivity. It can be viewed as the probability that a relevant document is retrieved by the query.

- Recall=TP/ (TP+FN)

F1-score:

- F1 score of the positive class in binary classification or weighted average of the F1 scores of each class for the multiclass task. When true positive + false positive == 0, precision is undefined. When true positive + false negative == 0, recall is undefined.
- F1-score= 2*((precision*recall)/ (precision+recall)).

3.5 System Architecture

SVM data mining technique has the highest accuracy across all conditions, and after using stepwise regression, 13 significant variables were screened and the classification accuracy of almost all data mining techniques was improved. However, the first 16 principal components transformed by PCA did not yield better classification results.

Therefore, the combination of SVM and the stepwise regression dimensionality reduction method.

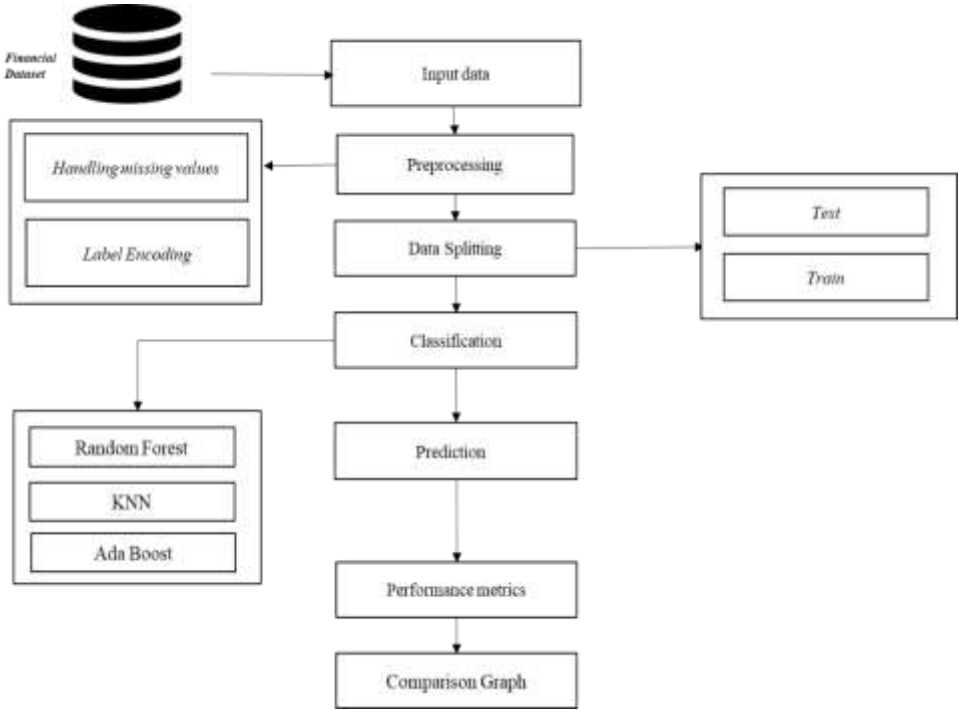


Fig.2:System Architecture

The proposed botnet detection model based on machine learning using DNS query data. The model is built on the analysis that Threats of CS Threats routinely send lookup queries to the DNS system to find IP addresses of C & C servers using automatically generated domain names.

The detection model is implemented in two phases: (a) the training phase and (b) the detection phase. During the training phase, the DNS query data is collected, and then domain names in DNS queries are extracted. Next, the set of domain names is pre-processed to extract the features for the training.

In the training phase, machine learning algorithms are used to learn the classifiers.Through the evaluation process, the machine learning algorithm that gives the highest overall classification accuracy will be selected for use in the proposed detection model. During the detection phase of the model, the DNS queries are monitored and passed through the process of extracting the domain names, pre-processing, and classifying using the classifier produced from the training phase to determine if a domain name is legitimate, or a Threats domain name.

The pre-processing step for each domain name in the training and detection phase is the same. However, this step is done in the offline mode for all the domain names of the training dataset in the training phase while it is done for each domain name extracted from the DNS query on the fly in the detection phase.

3.5.1 DETAILED DESCRIPTION

BLOCK DIGRAM

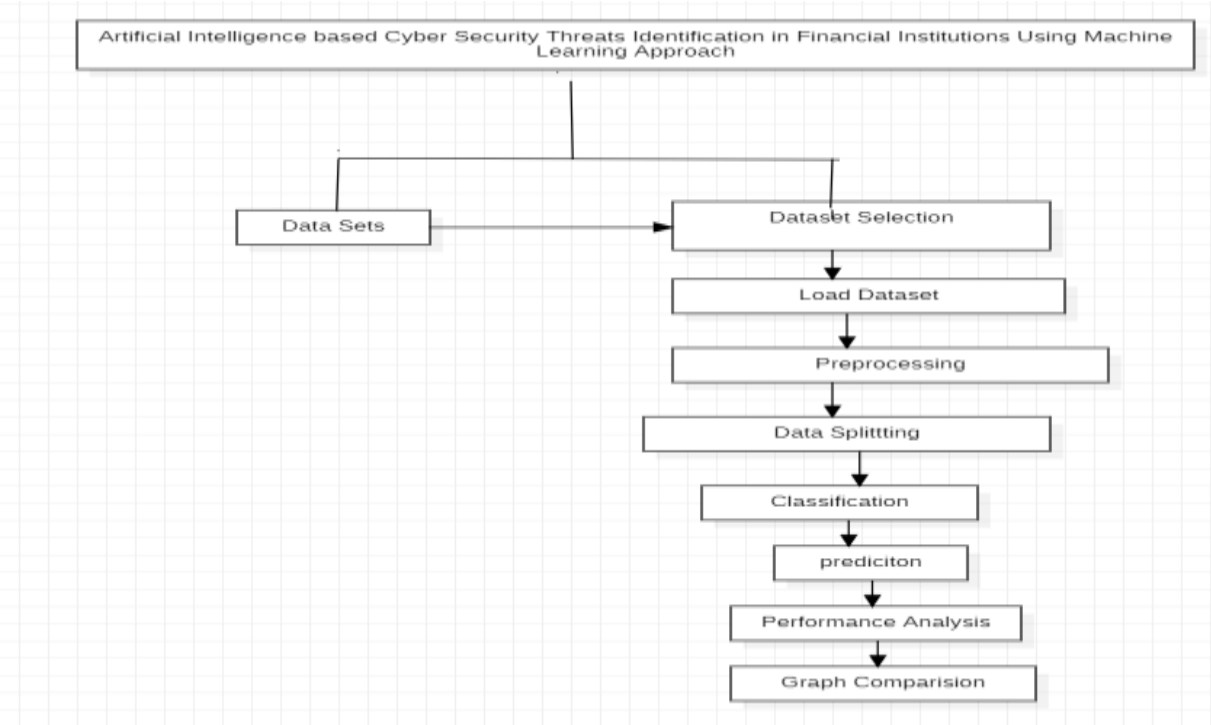


Fig.3:Block Diagram

This is the detailed project representation of our project

3.5.2 DATA FLOW DIAGRAM:

The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.

The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in

the system.DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.

DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

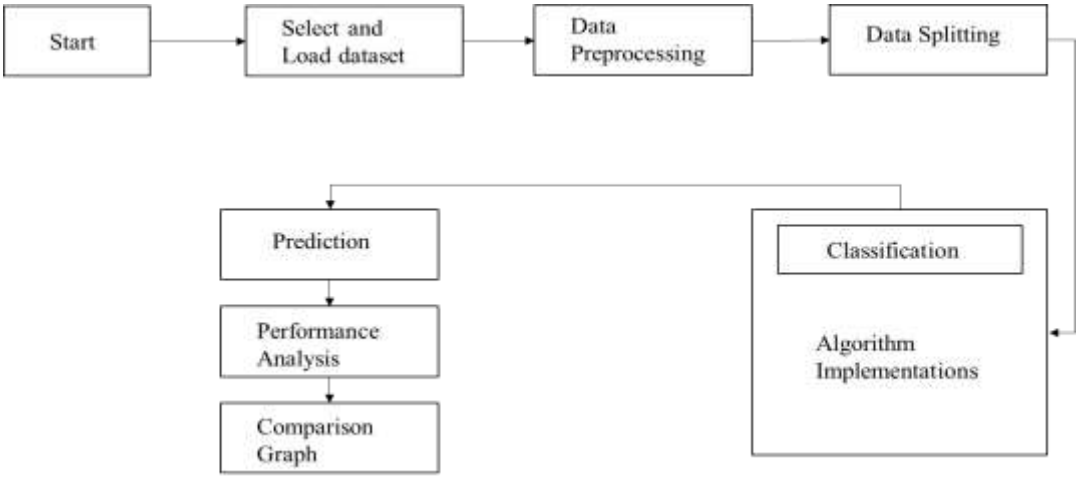


Fig.4:Data Flow Diagram

Four-step methodology was developed to identify existing studies in the literature that address Artificial Intelligence-based cyber-attacks. In addition, relevant information on the impact of attacks using AI is extracted to provide insights for structuring defense measures. The collection source was the Web of Science and Scopus database, covering the period between 2015 and 2022.

The database allows for retrieving a greater diversification of relevant metadata to the research. According to a systematic approach, the process of reviewing the literature was based on searching the following keywords: Artificial Intelligence, Machine Learning, Deep Learning, Cyber Security, Cybersecurity, and Industry 4.0. Although the literature review is not exhaustive, the method provides a comprehensive overview of the research topic in the literature.Steps of the Search Process:These databases, Web of Science and

Scopus, allow retrieving a greater diversification of relevant metadata to the research. In the Web of Science database with the field "TS = Topic" and the Scopus database with the field "TITLE-ABS". These tags combine fields that search document titles, abstracts, and keywords. The steps are described in the following:

Step 1—Identification: The keywords "Artificial Intelligence", "Machine Learning" and "Deep Learning" were combined with "Cyber Security", "Cybersecurity", and "Industry 4.0" in the advanced searches of the databases. The results of the searches are presented

Step 2—Screening:A filter excludes repeated publications. From a total of 219 publications, 81 repeated publications are identified, leaving a residual of 138 publications.

Step 3—Eligibility:A critical analysis evaluates the 138 selected publications. The goal is to filter out the studies that address the use of AI for both defense and cyber-attacks in the Industry 4.0 environment. In this step, 45 articles are identified after a filter is applied to exclude some selected document types: conference

papers, proceeding papers, review articles, books and chapters, early access, editorial material, show surveys, and not published in English. Altogether 93 documents are excluded from the search.

Step 4—Included:A critical reading of the material identified in step 3 is performed, considering the challenges and issues related to AI applied for cyber security in the context of Industry 4.0. After that, more than 18 studies were excluded, because they did not meet this criterion.

3.6 UML Diagrams

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

- The Primary goals in the design of the UML are as follows:
- Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
- Provide extendibility and specialization mechanisms to extend the core concepts.
- Be independent of particular programming languages and development process.
- Provide a formal basis for understanding the modeling language.

3.6.1 utilization Case Diagram

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

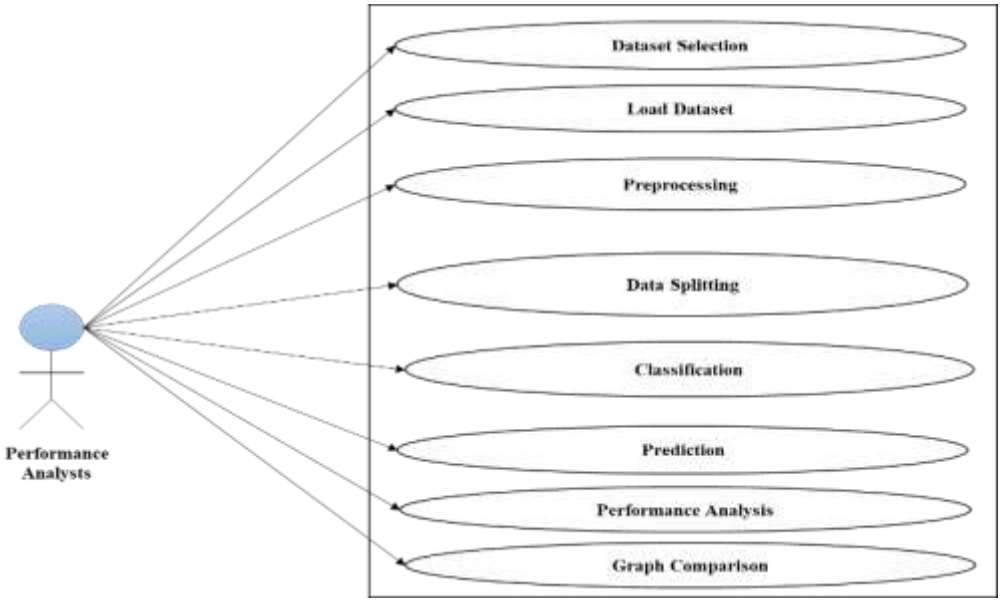


Fig.5:Utilization Case Diagram

3.6.2 State Diagram

A state diagram, as the name suggests, represents the different states that objects in the system undergo during their life cycle. Objects in the system change states in response to events. In addition to this, a state diagram also captures the transition of the object's state from an initial state to a final state in response to events affecting the system

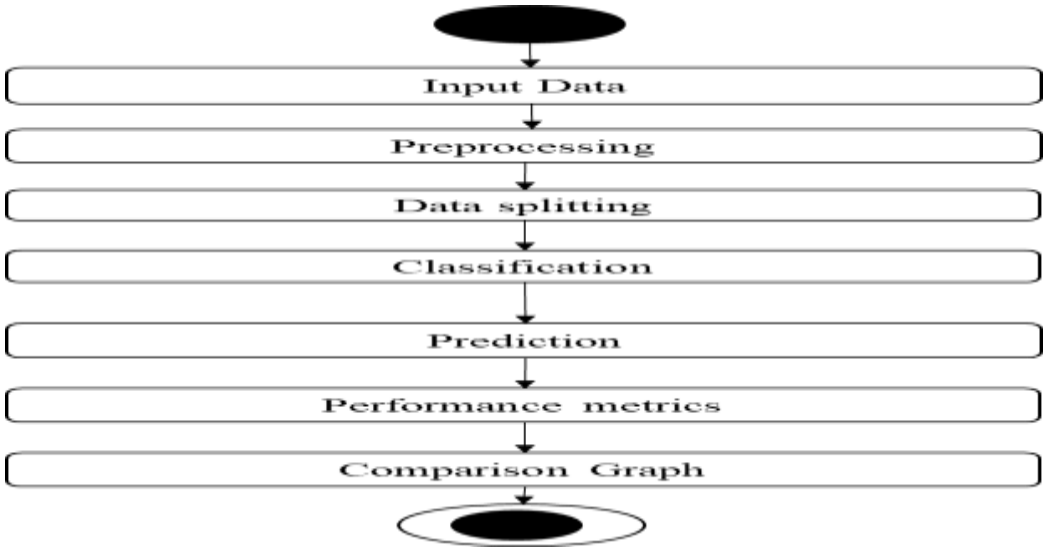


Fig.6:State Diagram

3.6.3 Sequence diagram

A sequence diagram represents the interaction between different objects in the system. The important aspect of a sequence diagram is that it is time-ordered. This means that the exact sequence of the interactions between the objects is represented step by step. Different objects in the sequence diagram interact with each other by passing "messages".

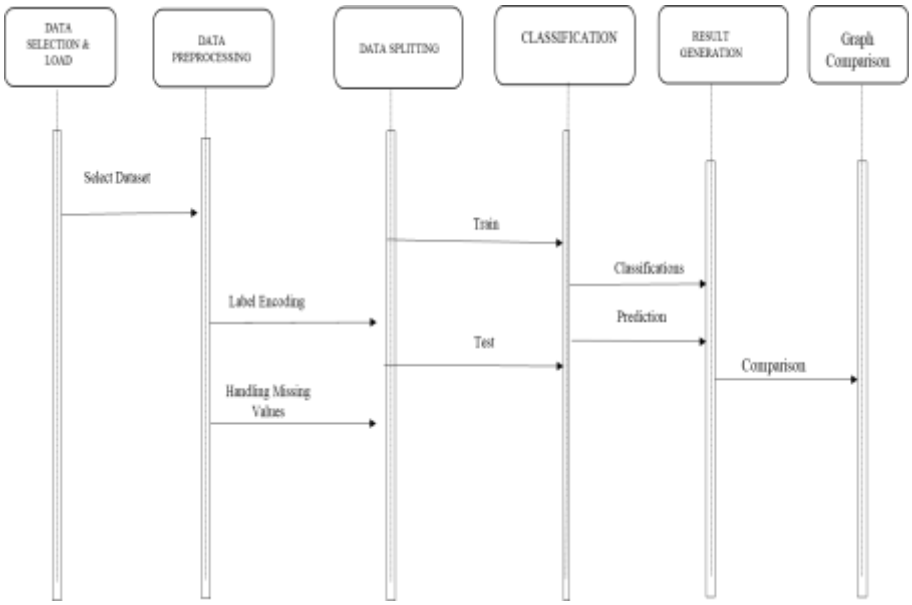


Fig.7:Sequence Diagram

3.6.4 Class Diagram

The class diagram is used to refine the use case diagram and define a detailed design of the system. The class diagram classifies the actors defined in the use case diagram into a set of interrelated classes. The relationship or association between the classes can be either an "is-a" or "has-a" relationship.

Each class in the class diagram may be capable of providing certain functionalities. These functionalities provided by the class are termed "methods" of the class. Apart from this, each class may have certain "attributes" that uniquely identify the class.

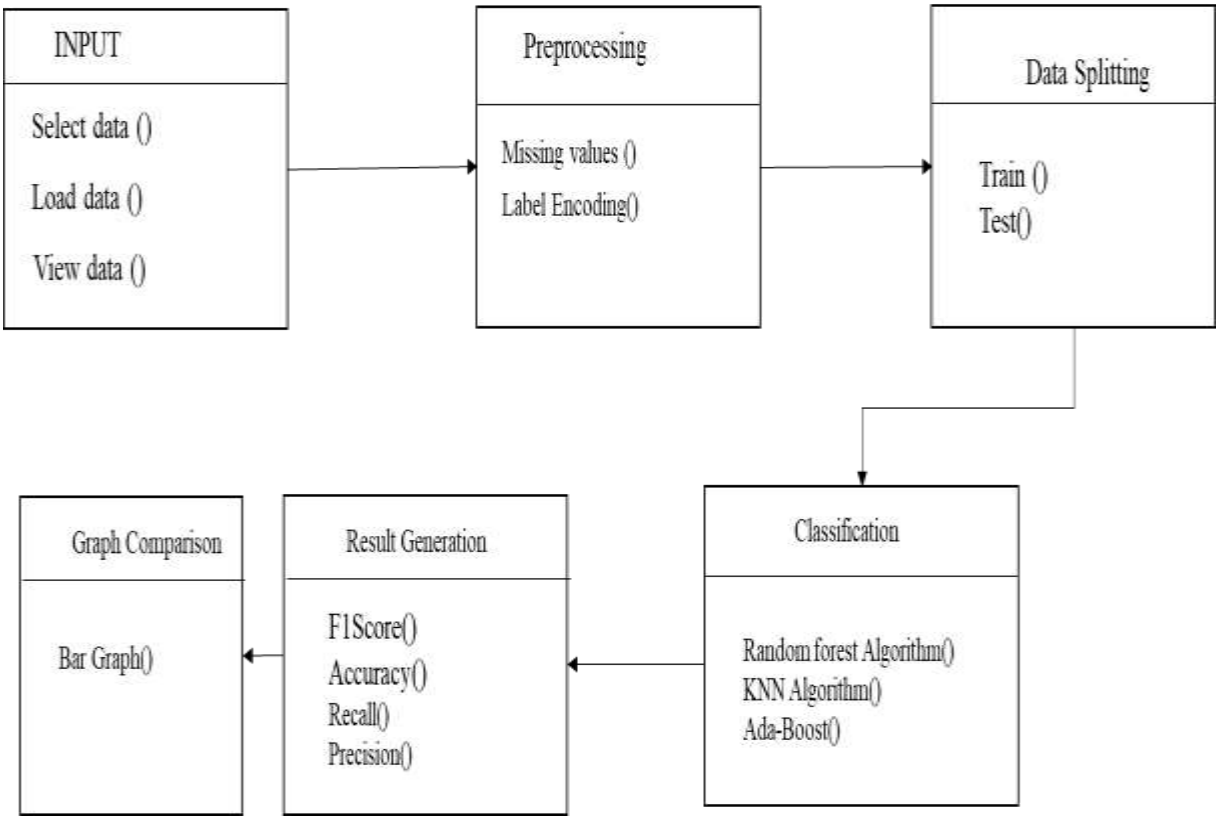


Fig.8:Class Diagram

3.6.5 ER DIAGRAM:

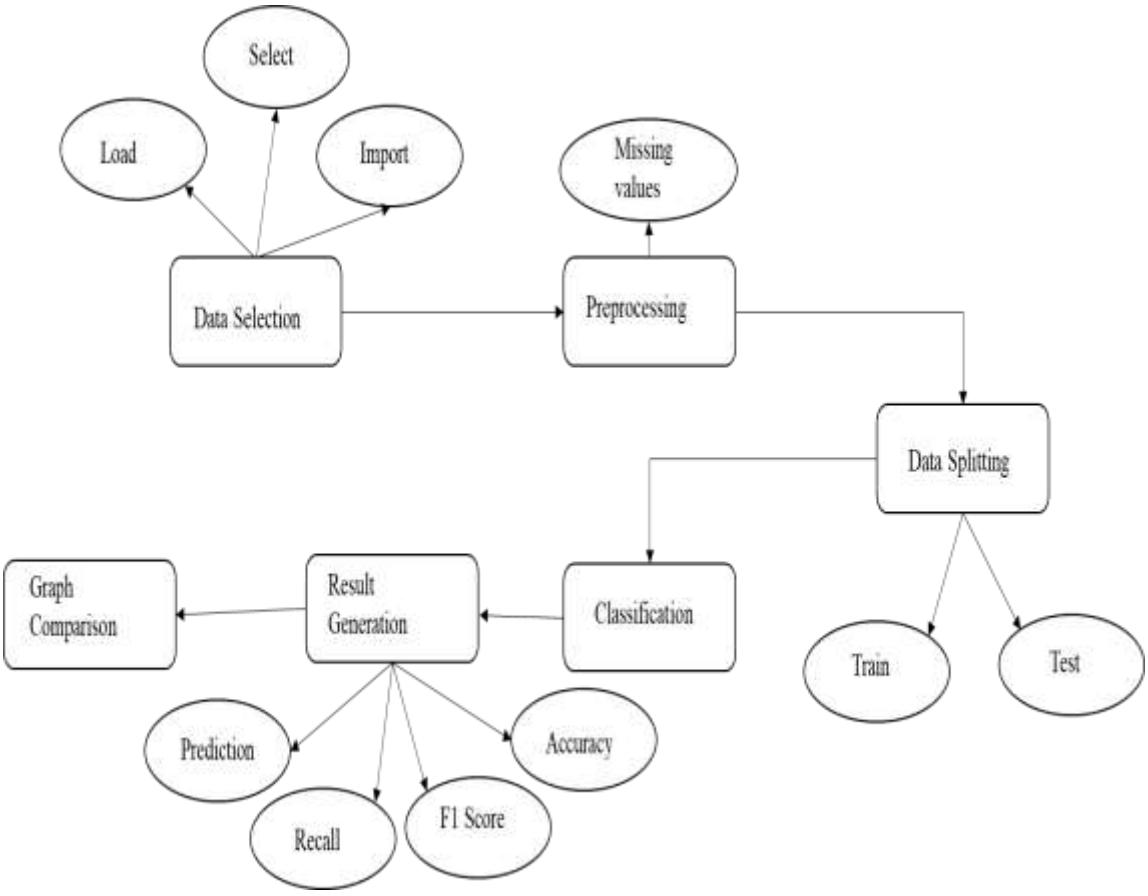


Fig.9:ER Diagram

3.7 Software and Hardware Requirements

3.7.1 Software Requirements

- Operating system : Windows 7 Ultimate.
- Coding Language : Python.
- Front-End : Python

3.7.2 Hardware Requirements

- System : Pentium IV 2.4 GHz.
- Hard Disk : 40 GB.
- Floppy Drive : 1.44 Mb.
- Monitor : 14’ Colour Monitor.
- Mouse : Optical Mouse.
- Ram : 512 Mb.

4.IMPLEMENTATION

4.1 Python

Python is one of those rare languages which can claim to be both simple and powerful. You will find yourself pleasantly surprised to see how easy it is to concentrate on the solution to the problem rather than the syntax and structure of the language you are programming in. The official introduction to Python is Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming.

Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms. I will discuss most of these features in more detail in the next section .

4.1.1 Features Of Python

Simple:

- Python is a simple and minimalistic language. Reading a good Python program feels almost like reading English, although very strict English! This pseudo-code nature of Python is one of its greatest strengths. It allows you to concentrate on the solution to the problem rather than the language itself.

Easy to Learn:

- As you will see, Python is extremely easy to get started with. Python has an extraordinarily simple syntax, as already mentioned.

Free and Open Source:

- Python is an example of a FLOSS (Free/Libre and Open Source Software). In simple terms, you can freely distribute copies of this software, read its source code, make changes to it, and use pieces of it in new free programs. FLOSS is based on the concept of a community which shares knowledge.
- This is one of the reasons why Python is so good - it has been created and is constantly improved by a community who just want to see a better Python.

High-level Language:

- When you write programs in Python, you never need to bother about the low-level details such as managing the memory used by your program, etc.

Portable:

- Due to its open-source nature, Python has been ported to (i.e. changed to make it
- work on) many platforms. All your Python programs can work on any of these platforms without requiring any changes at all if you are careful enough to avoid any system-dependent features.
- You can use Python on GNU/Linux, Windows, FreeBSD, Macintosh, Solaris, OS/2, Amiga, AROS, AS/400, BeOS, OS/390, z/OS, Palm OS, QNX, VMS, Psion, Acorn RISC OS, VxWorks, PlayStation, Sharp Zaru's, Windows CE and Pocke PC!
- You can even use a platform like Kavi to create games for your computer *and* for iPhone, iPad, and Android.

Interpreted:

- This requires a bit of explanation. A program written in a compiled language like C or C++ is converted from the source language i.e. C or C++ into a language that is spoken by your computer (binary code i.e. 0s and 1s) using a compiler with various flags and options. When you run the program, the linker/loader software copies the program from hard disk to memory and starts running it.
- Python, on the other hand, does not need compilation to binary. You just run the program directly from the source code. Internally, Python converts the source code into an intermediate form called bytecodes and then translates this into the native language of your computer and then runs it.
- All this, actually, makes using Python much easier since you don't have to worry about compiling the program, making sure that the proper libraries are linked and loaded, etc. This also makes your Python programs much more portable, since you can just copy your Python program onto another computer and it just works!

Object Oriented:

- Python supports procedure-oriented programming as well as object-oriented programming. In procedure-oriented languages, the program is built around procedures or functions which are nothing but reusable pieces of programs.
- In object-oriented languages, the program is built around objects which combine data and functionality. Python has a very powerful but simplistic way of doing OOP, especially when compared to big languages like C++ or Java.

Extensible:

- If you need a critical piece of code to run very fast or want to have some piece of algorithm not to be open, you can code that part of your program in C or C++ and then use it from your Python program.

Embeddable:

- You can embed Python within your C/C++ programs to give *scripting* capabilities for your program's users.

Extensive Libraries:

- The Python Standard Library is huge indeed. It can help you do various things involving regular expressions, documentation generation, unit testing, threading, databases, web browsers, CGI, FTP, email, XML, XML-RPC, HTML, WAV files, cryptography, GUI (graphical user interfaces), and other system-dependent stuff.
- Remember, all this is always available wherever Python is installed. This is called the Batteries Included philosophy of Python.
- Besides the standard library, there are various other high-quality libraries which you can find at the Python Package Index.

4.2 TECHNOLOGY DESCRIPTION

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such

as C++or Java.

It provides constructs that enable clear programming on both small and large scales. Python interpreters are available for many operating systems. C Python, the reference implementation of Python, is C Python is managed by the non-profit Python Software Foundation.

What is Python

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

- web development (server-side),
- software development,
- mathematics,
- system scripting.

What can Python do

- Python can be used on a server to create web applications.
- python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.

Why Python

- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
- Python has a simple syntax similar to the English language.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written This means that prototyping can be very quick.

Python Syntax compared to other programming languages:

- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

Introduction:

Python applications will often use packages and modules that don't come as part of the standard library. Applications will sometimes need a specific version of a library, because the application may require that a particular bug has been fixed or the application may be written using an obsolete version of the library's interface.

This means it may not be possible for one Python installation to meet the requirements of every application. If application A needs version 1.0 of a particular module but application B needs version 2.0, then the requirements are in conflict and installing either version 1.0 or 2.0 will leave one application unable to run.

The solution for this problem is to create a virtual environment, a self-contained directory tree that contains a Python installation for a particular version of Python, plus a number of additional packages. Different applications can then use different virtual environments.

To resolve the earlier example of conflicting requirements, application A can have its own virtual environment with version 1.0 installed while application B has another virtual environment with version 2.0. If application B requires a library be upgraded to version 3.0, this will not affect application A's environment.

4.2.1 Creating Virtual Environments

The module used to create and manage virtual environments is called vein. vein will usually install the most recent version of Python that you have available. If you have multiple versions of Python on your system, you can select a specific Python version by

running python3 or whichever version you want To create a virtual environment, decide upon a directory where you want to place it, and run the vein module as a script with the directory path:python3 -m vein tutorial-env This will create the tutorial-env directory if it doesn't exist, and also create directories inside it containing a copy of the Python interpreter, the standard library, and various supporting files.A common directory location for a virtual environment is very. A common directory location for a virtual environment is vein.

This name keeps the directory typically hidden in your shell and thus out of the way while giving it a name that explains why the directory exists.t also prevents clashing with. env environment variable definition files that some tooling supports. Once you've created a virtual environment, you may activate it.

On Windows, run:

tutorial-env\Scripts\activate.bat

On Unix or MacOS, run:

source tutorial-env/bin/activate

(This script is written for the bash shell. If you use the csh or fish shells, there are alternate activate.csh and activate.fish scripts you should use instead.)

Activating the virtual environment will change your shell's prompt to show what virtual environment you're using, and modify the environment so that running python will get you that particular version and installation of Python. For example:

```
$ source ~/envs/tutorial-env/bin/activate
(tutorial-env) $ python
Python 3.5.1 (default, May  6 2016, 10:59:36)
  ...
>>> import sys

>>>sys.path
```

['', '/usr/local/lib/python35.zip', ...,

'~/envs/tutorial-env/lib/python3.5/site-packages']

>>>

4.3 Managing Packages with pip

You can install, upgrade, and remove packages using a program called pip. By default pip will install packages from the Python Package Index, <https://pypi.org>. You can browse the Python Package Index by going to it in your web browser, or you can use pip's limited search feature:

(tutorial-env) $ pip search astronomy

sky field - Elegant astronomy for Python

Gary - Galactic astronomy and gravitational dynamics.

Novas - The United States Naval Observatory NOVAS astronomy library

Astros - Provides astronomy ephemeris to plan telescope observations

Astronomy - A collection of astronomy related tools for Python.

Modules guide for complete documentation for pip.)

You can install the latest version of a package by specifying a package's name:

(tutorial-env) $ pip install Novas

Collecting Novas

Downloading novas-3.1.1.3.tar.gz (136kB)

Installing collected packages: Novas

Running setup.py install for Novas

Successfully installed novas-3.1.1.3

You can also install a specific version of a package by giving the package name followed by == and the version number:

(tutorial-env) $ pip install requests==2.6.0

Collecting requests==2.6.0

Using cached requests-2.6.0-py2.py3-none-any.whl

Installing collected packages: requests

Successfully installed requests-2.6.0

If you re-run this command, pip will notice that the requested version is already installed and do nothing. You can supply a different version number to get that version, or you can run pip install --upgrade to upgrade the package to the latest version:

(tutorial-env) $ pip install --upgrade requests

Collecting requests

Installing collected packages: requests Found existing installation: requests 2.6.0

Uninstalling requests-2.6.0:

Successfully uninstalled requests-2.6.0

Successfully installed requests-2.7.0

pip uninstall followed by one or more package names will remove the packages from the virtual environment. Distribution taken from the supported dists parameter.

Deprecated since version 2.6.platform.linux_distribution(distname='', version='', id='', supported_dists=('SuSE', 'debian', 'redhat', 'mandrake', ...), full_distribution_name=1)Tries to determine the name of the Linux OS distribution name.

4.4 Using the Python Interpreter

4.4.1 Invoking the Interpreter

The Python interpreter is usually installed as /usr/local/bin/python3.8 on those machines where it is available; putting /usr/local/bin in your Unix shell's search path makes it possible to start it by typing the command:python3.8 to the shell. 1 Since the choice of the directory where the interpreter lives is an installation option, other places are possible; check with your local Python guru or system administrator. (E.g., /usr/local/python is a popular alternative location.)On Windows machines where you have installed Python from the Microsoft Store, the python3.8 command will be available.

If you have the py.exe launcher installed, you can use the py command. See Excursus: Setting environment variables for other ways to launch Python.

Typing an end-of-file character (Control-D on Unix, Control-Z on Windows) at the primary prompt causes the interpreter to exit with a zero exit status. If that doesn't work, you can exit the interpreter by typing the following command: quit().

The interpreter's line-editing features include interactive editing, history substitution and code completion on systems that support the GNU Read line library. Perhaps the quickest check to see whether command line editing is supported is typing Control-P to the first Python prompt you get. If it beeps, you have command line editing; see Appendix Interactive Input Editing and History Substitution for an introduction to the keys.

If nothing appears to happen, or if ^P is echoed, command line editing isn't available; you'll only be able to use backspace to remove characters from the current line.

The interpreter operates somewhat like the Unix shell: when called with standard input connected to a tty device, it reads and executes commands interactively; when called with a file name argument or with a file as standard input, it reads and executes a script from that file.

A second way of starting the interpreter is python -c command [arg] ..., which executes the statement(s) in command, analogous to the shell's -c option. Since Python statements often contain spaces or other characters that are special to the shell, it is usually advised to quote command in its entirety with single quotes. Python modules are also useful as scripts. These can be invoked using python -m module [arg] which executes the source file for module as if you had spelled out its full name on the command line.

When a script file is used, it is sometimes useful to be able to run the script and enter interactive mode afterwards. This can be done by passing -i before the script All command line options are described in Command line and environment Argument Passing.

When known to the interpreter, the script name and additional arguments thereafter are turned into a list of strings and assigned to the argv variable in the sys module. You can access this list by executing import sys. The length of the list is at least one; when no script and no arguments are given, sys.argv[0] is an empty string.

When the script name is given as '-' (meaning standard input), sys.argv[0] is set to '-'. When -c command is used, sys.argv[0] is set to '-c'. When -m module is used, sys.argv[0] is set to the full name of the located module. Options found after -c command or -m module are not consumed by the Python interpreter's option processing but left in sys.argv for the command or module to handle. Interactive Mode

When commands are read from a tty, the interpreter is said to be in interactive mode. In this mode it prompts for the next command with the primary prompt, usually three greater-than signs (>>>); for continuation lines it prompts with the secondary prompt, by default three dots (...). The interpreter prints a welcome message stating its version number and a copyright notice before printing the first prompt:

```
$ python3.8
Python 3.8 (default, Sep 16 2015, 09:25:04)
[GCC 4.8.2] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

4.5 The Interpreter and Its Environment

4.5.1 Source Code Encoding

By default, Python source files are treated as encoded in UTF-8. In that encoding, characters of most languages in the world can be used simultaneously in string literals, identifiers and comments although the standard library only uses ASCII characters for identifiers, a convention that any portable code should follow.

To display all these characters properly, your editor must recognize that the file is UTF-8, and it must use a font that supports all the characters in the file.To declare an encoding other than the default one, a special comment line should be added as the first line of the file. The syntax is as follows:

```
# -*- coding: encoding -*-
```

where encoding is one of the valid codecs supported by Python.

For example, to declare that Windows-1252 encoding is to be used, the first line of your source code file should be:

```
# -*- coding: cp1252 -*-
```

One exception to the first line rule is when the source code starts with a UNIX “shebang” line. In this case, the encoding declaration should be added as the second line of the file.

For example:

```
#!/usr/bin/env python3
```

```
# -*- coding: cp1252 -*-
```

4.6 Packages and Versions

```
astunparse          ==1.6.3
certifi             ==2022.12.7
charset-normalizer  ==3.0.1
centaury            ==1.0.7
cycler              ==0.11.0
Django              ==3.0.4
et-xmlfile          ==1.1.0
fonttools           ==4.38.0
gust                ==0.3.3
google-pasta        ==0.2.0
grpcio              ==1.51.1
h5py                ==2.10.0
Dina                ==3.4
importlib-metadata  ==6.0.0
joblib              ==1.2.0
Keres               ==2.3.1
Keras-Applications  ==1.0.8
```

5.SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable mKNNer. There are various types of test. Each test type addresses a specific testing requirement.

5.1 TYPES OF TESTS

Unit testing:

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration.

This is a structural testing, that relies on knowledge of its construction and is invalid Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing:

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields.

Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

- Valid Input : identified classes of valid input must be accepted.
- Invalid Input : identified classes of invalid input must be rejected.
- Functions : identified functions must be exercised.
- Output : identified classes of application outputs must be exercised.

Systems/Procedures:

Interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test:

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing:

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cKNNot be reached from a black box level.

Black Box Testing*:*

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document.It is a testing in which the software under test is treated, as a black box .you cKNNot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives:

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested:

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Integration Testing:

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

5.2 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

5.2.1 TESTCASES:

S.no	Test Case	Excepted Result	Result
1	Upload dataset	Dataset uploaded successfully	Pass
2	Preprocess data	Data preprocessing successfully	Pass
3	Handling Missing values	Handling Missing values generated successfully	Pass
4	Before Label Encoding	Successfully Encoding Before Label	Pass
5	After Label Encoding	Successfully Encoding After Label	Pass
6	Data splitting	Data splitting Successfully	Pass

7	Classification	Classification Successfully	Pass
8	Prediction	Prediction Successfully	Pass
9	Graph	Graph successfully	Pass

6.FUTURE ENHANCEMENT

In future, discovery of additional information based on cause-event Fraud detection well as prediction of detection based on cause events, etc. The working of the proposed approach in a web application.

7.CONCLUSION

In this project, we propose an approach to utilize the Random Forest algorithm, KNN and Ad boost algorithm for fraud detection in financial statements. We call the approach the three algorithms on datasets with significantly reduced dimensionality. The Classifications classifier gives high accuracy results that are comparable or superior to other fraud detection techniques in spite of working with reduced data and also compared with graph

SCREEN SHOTS

Data Selection:

```
#--------------------Data Selection---------------#
******************
```

	step	type	amount	...	newbalanceDest	isFraud	isFlaggedFraud
0	1	PAYMENT	NaN	...	0.00	0	0
1	1	PAYMENT	1864.28	...	0.00	0	0
2	1	TRANSFER	NaN	...	0.00	1	0
3	1	CASH_OUT	181.00	...	0.00	1	0
4	1	PAYMENT	11668.14	...	0.00	0	0
5	1	PAYMENT	7817.71	...	0.00	0	0
6	1	PAYMENT	7107.77	...	0.00	0	0
7	1	PAYMENT	7861.64	...	0.00	0	0
8	1	PAYMENT	4024.36	...	0.00	0	0
9	1	DEBIT	5337.77	...	40348.79	0	0
10	1	DEBIT	9644.94	...	157982.12	0	0
11	1	PAYMENT	3099.97	...	0.00	0	0
12	1	PAYMENT	2560.74	...	0.00	0	0
13	1	PAYMENT	11633.76	...	0.00	0	0
14	1	PAYMENT	4098.78	...	0.00	0	0
15	1	CASH_OUT	229133.94	...	51513.44	0	0
16	1	PAYMENT	1563.82	...	0.00	0	0
17	1	PAYMENT	1157.86	...	0.00	0	0
18	1	PAYMENT	671.64	...	0.00	0	0
19	1	TRANSFER	215310.30	...	0.00	0	0

Data Preprocessing:

Find Missing Values:

```
#--------------------Find missing values---------------#
******************
step               0
type               0
amount             2
nameOrig           0
oldbalanceOrg      0
newbalanceOrig     0
nameDest           0
oldbalanceDest     0
newbalanceDest     0
isFraud            0
isFlaggedFraud     0
dtype: int64
```

Handling Missing values:

```
#--------------------Fill 0 from missing Values---------------#
********************
step              0
type              0
amount            0
nameOrig          0
oldbalanceOrg     0
newbalanceOrig    0
nameDest          0
oldbalanceDest    0
newbalanceDest    0
isFraud           0
isFlaggedFraud    0
dtype: int64
```

Label Encoding:

```
#--------------------Before Label Encoding---------------#
********************
    step      type     amount  ... newbalanceDest  isFraud  isFlaggedFraud
0      1   PAYMENT       0.00  ...           0.00        0               0
1      1   PAYMENT    1864.28  ...           0.00        0               0
2      1  TRANSFER       0.00  ...           0.00        1               0
3      1  CASH_OUT     181.00  ...           0.00        1               0
4      1   PAYMENT   11668.14  ...           0.00        0               0
5      1   PAYMENT    7817.71  ...           0.00        0               0
6      1   PAYMENT    7107.77  ...           0.00        0               0
7      1   PAYMENT    7861.64  ...           0.00        0               0
8      1   PAYMENT    4024.36  ...           0.00        0               0
9      1     DEBIT    5337.77  ...       40348.79        0               0
10     1     DEBIT    9644.94  ...      157982.12        0               0
11     1   PAYMENT    3099.97  ...           0.00        0               0
12     1   PAYMENT    2560.74  ...           0.00        0               0
13     1   PAYMENT   11633.76  ...           0.00        0               0
14     1   PAYMENT    4098.78  ...           0.00        0               0
15     1  CASH_OUT  229133.94  ...       51513.44        0               0
16     1   PAYMENT    1563.82  ...           0.00        0               0
17     1   PAYMENT    1157.86  ...           0.00        0               0
18     1   PAYMENT     671.64  ...           0.00        0               0
19     1  TRANSFER  215310.30  ...           0.00        0               0
```

```
#--------------------After Label Encoding---------------#
********************
```

	step	type	amount	...	newbalanceDest	isFraud	isFlaggedFraud
0	1	3	0.00	...	0.00	0	0
1	1	3	1864.28	...	0.00	0	0
2	1	4	0.00	...	0.00	1	0
3	1	1	181.00	...	0.00	1	0
4	1	3	11668.14	...	0.00	0	0
5	1	3	7817.71	...	0.00	0	0
6	1	3	7107.77	...	0.00	0	0
7	1	3	7861.64	...	0.00	0	0
8	1	3	4024.36	...	0.00	0	0
9	1	2	5337.77	...	40348.79	0	0
10	1	2	9644.94	...	157982.12	0	0
11	1	3	3099.97	...	0.00	0	0
12	1	3	2560.74	...	0.00	0	0
13	1	3	11633.76	...	0.00	0	0
14	1	3	4098.78	...	0.00	0	0
15	1	1	229133.94	...	51513.44	0	0
16	1	3	1563.82	...	0.00	0	0
17	1	3	1157.86	...	0.00	0	0
18	1	3	671.64	...	0.00	0	0
19	1	4	215310.30	...	0.00	0	0

Data Splitting:

```
#-------------------Data Splitting--------------#
********************

Total no of dataset : (80000, 11)
Training set Without Target (64000, 10)
Training set only Target (64000,)
Testing set Without Target (16000, 10)
Testing set only Target (16000,)
```

Classification:

```
#--------------------Random Forest Algorithm---------------#
********************
Matrix:
[[15976     0]
 [   12    12]]
classfication:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     15976
           1       1.00      0.50      0.67        24

   micro avg       1.00      1.00      1.00     16000
   macro avg       1.00      0.75      0.83     16000
weighted avg       1.00      1.00      1.00     16000

Accuracy:  99.925
```

```
#--------------------KNN Algorithm---------------#
********************

Matrix:
[[15975     1]
 [   24     0]]
classfication:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     15976
           1       0.00      0.00      0.00        24

   micro avg       1.00      1.00      1.00     16000
   macro avg       0.50      0.50      0.50     16000
weighted avg       1.00      1.00      1.00     16000

Accuracy:  99.84375
```

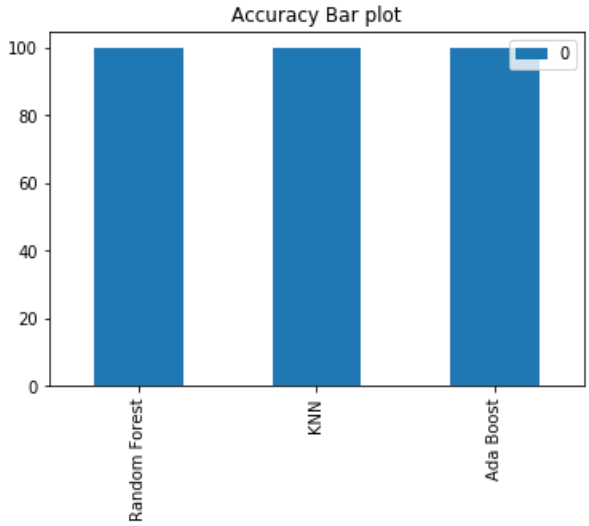
```
#--------------------Ada Boost---------------#
********************
0.999

Matrix:
[[15973     3]
 [   13    11]]
classfication:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     15976
           1       0.79      0.46      0.58        24

   micro avg       1.00      1.00      1.00     16000
   macro avg       0.89      0.73      0.79     16000
weighted avg       1.00      1.00      1.00     16000

Accuracy:  99.9
```

Prediction:

```
#--------------------Get input from user---------------#
********************

Enter the Step: 1

Enter the Type: 4

Enter the Amount: 0

Enter the nameOrig: 15121

Enter the oldbalance: 181

Enter the newbalance: 0

Enter the nameDest: 7874

Enter the oldbalance: 0

Enter the newbalance: 0

Enter the isFlaggedFraud: 0
[1]
This is financial Fraud
```

GRAPH:

```
#--------------------Camparision batween 3 Algorithm Accuracy---------------#
********************
```



REFERENCES

- Albizri, D. Appelbaum, and N. Rizzotto, ‘‘Evaluation of financial statements fraud detection research: A multi-disciplinary analysis,’’ Int. J. Discl. Governance, vol. 16, no. 4, pp. 206–241, Dec. 2019.
- R.Albright, ‘‘Taming text with the SVD.SAS institute white paper, ’’SAS Inst., Cary, NC, USA, White Paper 10.1.1.395.4666, 2004.
- M. S. Beasley, ‘‘An empirical analysis of the relation between the board of director composition and financial statement fraud,’’ Accounting Rev., vol. 71, pp. 443–465, Oct. 1996.
- T. B. Bell and J. V. Carcello, ‘‘A decision aid for assessing the likelihood of fraudulent financial reporting,’’ Auditing A, J. Pract. Theory, vol. 19, no. 1, pp. 169–184, Mar. 2000.
- M.D.BeneishandC.Nichols,‘‘The predictable cost of earnings manipulation,’’Dept.Accounting,KelleySchoolBus.,IndianaUniv.,Bloomington, IN, USA, Tech. Rep. 1006840, 2007.
- R. J. Bolton and D. J. Hand, ‘‘Statistical fraud detection: A review,’’ Stat. Sci., vol. 17, no. 3, pp. 235–249, Aug. 2002.
- M.Cecchini, H.Aytug, G.J.Koehler, and P.Pathak,‘‘Making words work: Using financial text as a predictor of financial events,’’ Decis. Support Syst., vol. 50, no. 1, pp. 164–175, 2010.
- Q. Deng, ‘‘Detection of fraudulent financial statements based on naïve Bayes classifier,’’ in Proc. 5th Int. Conf. Comput. Sci. Educ., 2010, pp. 1032–1035.
- S. Chen, Y.-J.-J. Goo, and Z.-D. Shen, ‘‘A hybrid approach of stepwise regression, logistic regression, support vector machine, and decision tree for forecasting fraudulent financial statements,’’ Sci. World J., vol. 2014, pp. 1–9, Aug. 2014. X. Chen and R. Ye, ‘‘Identification model of logistic regression analysis on listed Firms’ frauds in China,’’ in Proc. 2nd Int. Workshop Knowl. Discovery Data Mining, Jan. 2009, pp. 385–388.
- R. Cressey, ‘‘Other people’s money; a study of the social psychology of embezzlement,’’ Amer. J. Sociol.,
- vol. 59, no. 6, May 1954, doi: 10.1086/221475.

Creative Soft
Development, Consultancy & Out-Sourcing

PROJECT COMPLETION CERTIFICATE

This is to conform that, Mr. Singa SanjeevaRaju Studying MCA bearing the Reg. No: 22091F0047 from “Rajeev Gandhi Memorial College of Engineering and Technology, NANDYAL”has successfully completed his project work titled “AI-POWERED CYBERSECURITY: MACHINE LEARNING FOR THREAT IDENTIFICATION IN FINANCIAL INSTITUTIONS” on Python Technologies as part of her course Curriculum. He has done this project using Python during the period 01-Feb-2024 to19-May-2024 under the guidance and supervision of Mr. Sudharshan Reddy Kota from “CREATIVE SOFT SOLUTIONS”, Hyderabad.

He has completed the assigned project well within the time frame. He is sincere, hardworking and him conduct during the project is commendable.

For Creative Soft

Address: #510, Annapurna Block, beside mytrivanam, Adhithya Enclave, Ameerpet, Hyd-38.Cell: 9502354142

(ESTD-1995)

Rajeev Gandhi Memorial College of Engineering and Technology (RGM), Andhra Pradesh

Certificate of Plagiarism Check for Thesis

Author Name	SINGA SANJEEVARAJU (22091F0047)
Course of Study	**MCA IV Sem. (R20)- May-2024**
Name of Guide	
Department	MCA
Acceptable Maximum Limit	30%
Submitted By	principal.9@jntua.ac.in
Paper Title	AI POWERED CYBERSECURITY MACHINE LEARNING FOR THREAT IDENTIFICATION IN FINANCIAL INSTITUTIONS
Similarity	21%
Paper ID	1868007
Submission Date	2024-05-25 13:32:58

Note: Less than 30% Similarity index – Permitted to Submit the Thes

Controller of Examinations

* This report has been generated by DrillBit Anti-Plagiarism Software

DrillBit

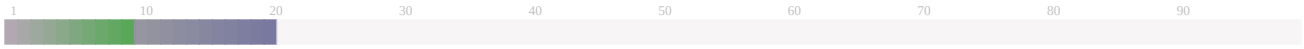
The Report is Generated by DrillBit Plagiarism Detection Software

Submission Information

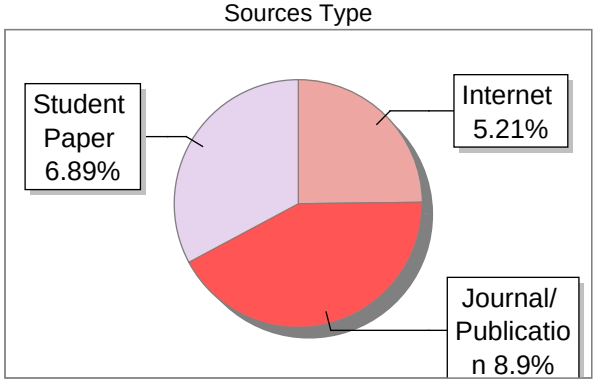
Author Name	SINGA SANJEEVARAJU (22091F0047)
Title	AI POWERED CYBERSECURITY MACHINE LEARNING FOR THREAT IDENTIFICATION IN FINANCIAL INSTITUTIONS
Paper/Submission ID	1868007
Submitted by	principal.9@jntua.ac.in
Submission Date	2024-05-25 13:32:58
Total Pages, Total Words	31, 6628
Document type	Project Work

Result Information

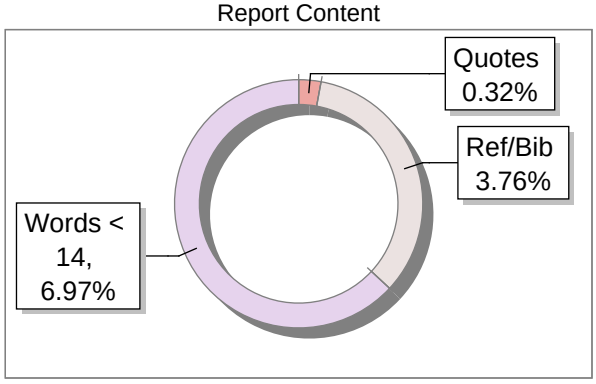
Similarity **21 %**



Sources Type



Report Content



Exclude Information

Quotes	Not Excluded
References/Bibliography	Not Excluded
Source: Excluded < 14 Words	Not Excluded
Excluded Source	**0 %**
Excluded Phrases	Not Excluded

Database Selection

Language	English
Student Papers	Yes
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	Yes

A Unique QR Code use to View/Download/Share Pdf File

DrillBit

DrillBit Similarity Report

21	63	B
SIMILARITY %	MATCHED SOURCES	GRADE

A-Satisfactory (0-10%)
B-Upgrade (11-40%)
C-Poor (41-60%)
D-Unacceptable (61-100%)

LOCATION	MATCHED DOMAIN	%	SOURCE TYPE
1	sist.sathyabama.ac.in	3	Publication
2	REPOSITORY - Submitted to Jawaharlal Nehru Technological University (H) on 2024-05-08 15-27 1000661	1	Student Paper
3	Submitted to Visvesvaraya Technological University, Belagavi	1	Student Paper
4	www.mdpi.com	1	Internet Data
5	naac.mictech.edu.in	1	Publication
6	Submitted to Visvesvaraya Technological University, Belagavi	1	Student Paper
7	PMOD SECURE PRIVILEGE BASED MULTILEVEL ORGANIZATIONAL DATA SHARING IN CLOUD COMPUTING BY 18S41F0026 Yr-2021 SUBMITTED TO JNTU	1	Student Paper
8	thesai.org	1	Publication
9	REPOSITORY - Submitted to Jawaharlal Nehru Technological University (H) on 2024-05-08 17-51 675166	<1	Student Paper
10	Submitted to Visvesvaraya Technological University, Belagavi	<1	Student Paper
11	pec.paavai.edu.in	<1	Publication
12	Submitted to Visvesvaraya Technological University, Belagavi	<1	Student Paper

13	Submitted to Visvesvaraya Technological University, Belagavi	<1	Student Paper
14	www.science.gov	<1	Internet Data
15	www.ijcrt.org	<1	Publication
16	archive.mu.ac.in	<1	Publication
17	www.scribd.com	<1	Internet Data
18	Submitted to Visvesvaraya Technological University, Belagavi	<1	Student Paper
19	www.ncbi.nlm.nih.gov	<1	Internet Data
20	REPOSITORY - Submitted to Jawaharlal Nehru Technological University (H) on 2024-05-08 17-51 647125	<1	Student Paper
21	Selection of optimized features for fusion of palm print and finger knucklebase by Jaswal-2020	<1	Publication
22	Theoretical consideration and modeling of self-healing polymers by Min-2012	<1	Publication
23	blog.devart.com	<1	Internet Data
24	dspace.daffodilvarsity.edu.bd 8080	<1	Publication
25	moam.info	<1	Internet Data
26	Submitted to Visvesvaraya Technological University, Belagavi	<1	Student Paper
27	naac.mictech.edu.in	<1	Publication
28	www.mdpi.com	<1	Publication
29	baeldung3.rssing.com	<1	Internet Data
30	index-of.es	<1	Publication

31	Thesis Submitted to Shodhganga, shodhganga.inflibnet.ac.in	<1	Publication
32	www.dx.doi.org	<1	Publication
33	www.kansascityfed.org	<1	Publication
34	1library.co	<1	Internet Data
35	A new heuristic for capturing the complexity of multimodal signals by Smith-2013	<1	Publication
36	bramblegarden.com	<1	Internet Data
37	digitalscholarship.unlv.edu	<1	Internet Data
38	REPOSITORY - Submitted to Jawaharlal Nehru Technological University (H) on 2024-05-07 16-54 1617443	<1	Student Paper
39	REPOSITORY - Submitted to Jawaharlal Nehru Technological University (H) on 2024-05-08 17-51 666332	<1	Student Paper
40	Submitted to Visvesvaraya Technological University, Belagavi	<1	Student Paper
41	vincechiarelli.com	<1	Internet Data
42	Thermal infrared detection of cavities in trees by G-1990	<1	Publication
43	worldwidescience.org	<1	Internet Data
44	ag.ny.gov	<1	Publication
45	Atomisticcontinuum interphase model for effective properties of composite mater by Paliwal-2011	<1	Publication
46	CUSTOMER CHURN ANALYSIS AND PREDICTION USING DATA MINING MODEL IN BANKING INDUSTRY BY 15W71D5803 Yr-2021 SUBMITTED TO JNTU	<1	Student Paper

47	Dissimulation or Assimilation The Case of the Mandans 1, by Hberl, Charles G.- 2013	<1	Publication
48	edkentmedia.com	<1	Internet Data
49	qdoc.tips	<1	Internet Data
50	1library.co	<1	Internet Data
51	academic.oup.com	<1	Internet Data
52	fjfsdata01prod.blob.core.windows.net	<1	Publication
53	INDUSTRIAL LITERATURE by -1943	<1	Publication
54	moam.info	<1	Internet Data
55	Submitted to Visvesvaraya Technological University, Belagavi	<1	Student Paper
56	Submitted to Visvesvaraya Technological University, Belagavi	<1	Student Paper
57	technodocbox.com	<1	Internet Data
58	Thesis submitted to dspace.mit.edu	<1	Publication
59	www.biorxiv.org	<1	Internet Data
60	www.datasciencecentral.com	<1	Internet Data
61	www.readbag.com	<1	Internet Data
62	www.science.gov	<1	Internet Data
63	www.shopify.com	<1	Internet Data

Strad Research

An UGC-CARE Approved Group - 2 Journal
An ISO: 7021 - 2008 Certified Journal
ISSN: 0039-2049, Website:http://stradresearch.org/
email: editorstrad@gmail.com

CERTIFICATE OF PUBLICATION

This is to certify that the paper entitled

AI-POWERED CYBERSECURITY: MACHINE LEARNING FOR THREAT IDENTIFICATION IN FINANCIAL INSTITUTIONS

Authored by

Mrs.S TAHSEEN BANU

From

Rajeev Gandhi Memorial College of Engineering and Technology, Nandyal, Andhra Pradesh

Has been published in

STRAD RESEARCH, VOLUME 11, ISSUE 5, MAY – 2024.

J Palis

Editor-in-Chief,

Strad Research

An UGC-CARE Approved Group - 2 Journal
An ISO: 7021 - 2008 Certified Journal
ISSN: 0039-2049, Website:http://stradresearch.org/
email: editorstrad@gmail.com

CERTIFICATE OF PUBLICATION

This is to certify that the paper entitled

AI-POWERED CYBERSECURITY: MACHINE LEARNING FOR THREAT IDENTIFICATION IN FINANCIAL INSTITUTIONS

Authored by

SINGA SANJEEVARAJU

From

Rajeev Gandhi Memorial College of Engineering and Technology, Nandyal, Andhra Pradesh

Has been published in

STRAD RESEARCH, VOLUME 11, ISSUE 5, MAY – 2024.

J Palis

Editor-in-Chief,

AI-POWERED CYBERSECURITY: MACHINE LEARNING FOR THREAT IDENTIFICATION IN FINANCIAL INSTITUTIONS

[1]Mrs.S TAHSEEN BANU, [2]SINGA SANJEEVARAJU
[1]Assistant Professor, [2]MCA Student
Department of Master of Computer application,
Rajeev Gandhi Memorial College of Engineering and Technology
Nandyal, 518501, Andhra Pradesh, India.

ABSTRACT

As digital assets become more interconnected, cyber threats are growing at an unprecedented rate. Financial institutions need to invest in artificial intelligence-based solutions for identifying these threats and protecting their assets. Machine learning is a powerful tool for investigating complex financial security threats that constantly evolve and can be difficult to predict. By leveraging AI technologies such as natural language processing, algorithms, and automated reasoning systems, banks can develop a better understanding of potential risks and create more efficient controls around their data. In this paper, an artificial intelligence based cyber security threats identification has proposed in financial institutions using machine learning approach. Machine learning algorithms are constantly being improved to identify anomalies in the data that might indicate a security threat. This approach enables financial firms to identify and defend against malicious attacks using custom-made models that provide actionable insights into both internal and external risks.

1. INTRODUCTION

Financial fraud refers to the use of fraudulent and illegal methods or deceptive tactics to gain financial benefits. Fraud can be committed in different areas of finance, including banking, insurance, taxation, and corporates, and more. Fiscal fraud and evasion, including credit card fraud, tax evasion, financial statement fraud, money laundry, and other types of financial fraud, has become a growing problem. Despite efforts to eliminate financial fraud, its occurrence adversely affects business and society as hundreds of millions of dollars are lost to fraud each year. This significant financial loss has dramatically affected individuals, merchants, and banks.

Nowadays, fraud attempts have increased drastically, which makes fraud detection more important than ever. The Association of Certified Fraud Examiners (ACFE) has announced that 10% of incidents concerning white-collar crime involves falsification of financial statements. They classified occupational fraud into three types: asset misappropriation, corruption, and financial statement fraud. Financial statement fraud resulted in the most significant losses among them.

Although the occurrence frequency of asset misappropriation and corruption is much higher than financial statement fraud, the financial implications of these latter crimes are still far less severe. In particular, as reported in a survey from Eisner Amper, which is among the prominent accounting firms in the U.S., ‘‘the average median loss of financial statement fraud ($800,000 in 2018) accounts for over three times the monetary loss of corruption ($250,000) and seven times as much as asset misappropriation ($114,000)’’.

The focus of this study is on financial statement fraud. Financial statements are documents that describe details about a company, specifically their business activities and financial performance, including income, expenses, profits, loans, presumable concerns that may emerge later, and managerial comments on the business performance.

All firms are obligated to announce their financial statements in a quarterly and annual manner. Financial statements can be used to indicate the performance of a company. Investors, market analysts, and creditors exploit financial reports to investigate and assess the financial health and earnings potentials of a business. Financial statements consist off our sections; income statement, balance sheet, cash flow statement, and explanatory notes. The income statement places a great emphasis on a company’s expenses and revenues during a specific period.

The company’s profit or net income is provided in this section, which subtracts expenses from revenues. The balance sheet provides a timely snapshot of liabilities, assets, and stockholders’ equity. The cash flow statement measures the extent to which a company is successful in making cash to fund its operating expenses, fund investments, and pay its debt obligations. Explanatory notes are supplemental data that provide clarification and further information about particular items published financial statements of a company.

These notes cover areas including disclosure of subsequent events, asset depreciation, and significant accounting policies, which are necessary disclosures that demonstrate the amounts reported on the financial statements. Financial statement fraud involves falsifying financial statements to pretend the company more profitable than it is, increase the stock prices, avoid payment of the taxes, or get a bank loan.

Fraud triangle in auditing is a framework to demonstrate the motivation behind an individual’s decision to commit fraud. The fraud triangle has three elements that increase the risk of fraud: incentive, rationalization, and opportunity, which, together, lead to fraudulent behavior. Auditing professionals have extensively used this theory to explain the motivation behind an individual’s decision to commit fraud.

It is indispensable to understand the fraud triangle to evaluate financial fraud. Gupta and Singh suggested that when there are incentives such as the obligation to achieve an outcome or cover losses, the

potential for fraud increases. The company will encounter temptations or pressures to adopt fraudulent practices.

Moreover, the lack of inspections or unsuccessful controls provides a favourable occasion for committing fraud. Rationalization happens when the fraudster aims to justify the fraudulent action, and it could be affected by the others and the conditions.

2. LITERATURE SURVEY

Evaluation of financial statements fraud detection research: A multidisciplinary analysis

Prior research in the fields of accounting and information systems has shed some light on the significant effects of financial reporting fraud on multiple levels of the economy. In this paper, we compile prior multi-disciplinary literature on financial statement fraud detection. Financial reporting fraud detection efforts and research may be more impactful when the findings of these different domains are combined. We anticipate that this research will be valuable for academics, analysts, regulators, practitioners, and investors.

Interpretable fuzzy rule-based systems for detecting financial statement fraud

Systems for detecting financial statement frauds have attracted considerable interest in computational intelligence research. Diverse classification methods have been employed to perform automatic detection of fraudulent companies. However, previous research has aimed to develop highly accurate detection systems, while neglecting the interpretability of those systems. Here we propose a novel fuzzy rule-based detection system that integrates a feature selection component and rule extraction to achieve a highly interpretable system in terms of rule complexity and granularity. Specifically, we use a genetic feature selection to remove irrelevant attributes and then we perform a comparative analysis of state-of-the-art fuzzy rule-based systems, including FURIA and evolutionary fuzzy rule-based systems. Here, we show that using such systems leads not only to competitive accuracy but also to desirable interpretability. This finding has important implications for auditors and other users of the detection systems of financial statement fraud.

An application of ensemble random forest classifier for detecting financial statement manipulation of Indian listed companies

A rising incidents of financial frauds in recent time has increased the risk of investor and other stakeholders. Hiding of financial losses through fraud or manipulation in reporting and hence resulted into erosion of considerable wealth of their stakeholders. In fact, a number of global companies like WorldCom, Xerox, Enron and number Indian companies such as Satyam, Kingfisher and Deccan Chronicle had committed fraud in financial statement by manipulation. Hence, it is imperative to create an efficient and effective framework for detection of financial fraud. This can be helpful to regulators, investors, governments and auditors as preventive steps in avoiding any possible financial fraud cases. In this context, increasing number of researchers

these days have started focusing on developing systems, models and practices to detect fraud in early stage to avoid the any attrition of investor's wealth and to reduces the risk of financing.

In Current study, the researcher has attempted to explore the various 42 modeling techniques to detect fraud in financial statements (FFS). To perform the experiment, researcher has chosen 86 FFS and 92 non-fraudulent financial statements (nonFFS) of manufacturing firms. The data were taken from Bombay Stock Exchange for the dimension of 2008-2011. Auditor's report is considered for classification of FFS and Non-FFS companies. T-test was applied on 31 important financial ratios and 10 significant variables were taken in to consideration for data mining techniques. 86 FFS and 92 non-FFS during 2008-2017 were taken for testing data set. Researcher has trained the model using data sets. Then, the trained model was applied to the testing data set for the accuracy check. Random forest gives best accuracy. Here, modified random forest model was developed with improved accuracy.

3. EXISTING SYSTEM:

Fraudulent financial statements (FFS) are the results of manipulating financial elements by overvaluing incomes, assets, sales, and profits while underrating expenses, debts, or losses. To identify such fraudulent statements, traditional methods, including manual auditing and inspections, are costly, imprecise, and time-consuming. Intelligent methods can significantly help auditors in analyzing a large number of financial statements. In this study, we systematically review and synthesize the existing literature on intelligent fraud detection in corporate financial statements. In particular, the focus of this review is on exploring machine learning and data mining methods, as well as the various datasets that are studied for detecting financial fraud. We adopted the Kitchen ham methodology as a well-defined protocol to extract, synthesize, and report the results. Accordingly, 47 articles were selected, synthesized, and analyzed. We present the key issues, gaps, and limitations in the area of fraud detection in financial statements and suggest areas for future research. Since supervised algorithms were employed more than unsupervised approaches like clustering, the future research should focus on unsupervised, semi-supervised, as well as bio-inspired and evolutionary heuristic methods for fraud (fraud) detection. In terms of datasets, it is envisaged that future research making use of textual and audio data. While imposing new challenges, this unstructured data deserves further study as it can show interesting results for intelligent fraud detection.

DISADVANTAGES:

- The results is low when compared with proposed.
- Time consumption is high.
- Theoretical limits.

4. PROPOSED SYSTEM

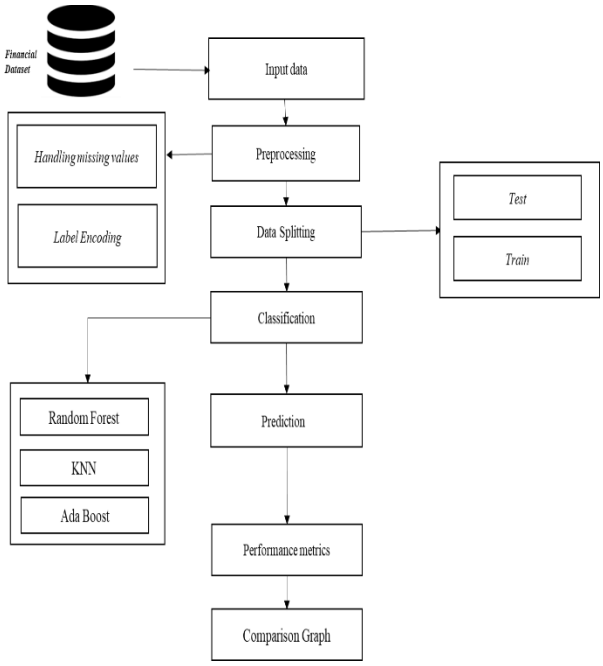
In our proposed system, we detect the fraud in financial statements by using the machine learning algorithm. First, we select and view the imported dataset for future purpose. And we get missing values and fill the default values to the dataset. We encoding the label in the dataset. And we split the dataset to the

Train and Test data for predict the fraud or non-fraud. Then we use three algorithms for more accuracy, prediction and which is more accurate value. There are Random forest algorithm, KNN classifiers and Ada-Boost Algorithm. Now, we fit the training data from the dataset. Then we predict the test dataset using training dataset. Then the test values get the results of actual and predicted. And we get the performance of the dataset. It is essential to train the models on data which includes fraud and relevant non fraud. By using the ML algorithm the system is, to classify the fraud and non-fraud and results shows that the accuracy, precision, recall and f1-score and also prediction. This shows that method used in this project can predict the possibility of fraud accurately in most of the cases. This module is the simple and effective way to avoid such frauds and save those expenditures.

ADVANTAGES

- It is efficient for large number of datasets.
- The experimental result is high when compared with existing system.
- Time consumption is low.
- Provide accurate prediction results.

5. SYSTEM ARCHITECTURE



6. IMPLEMENTATION

MODULES DESCRIPTION:

DATA SELECTION:

- The input data was collected from the dataset repository like UCI Repository.
- In this process, the input data have some columns like step, type, amount, nameOrig, balanceOrig, nameDest, balanceDest, isFlaggedFraud, etc.

In our collected dataset was read in this process using pandas.

DATA PREPROCESSING:

- Data pre-processing is the process of removing the unwanted data from the dataset.

- Pre-processing data transformation operations are used to transform the dataset into a structure suitable for machine learning.
- This step also includes cleaning the dataset by removing irrelevant or corrupted data that can affect the accuracy of the dataset, which makes it more efficient.
- Missing data removal
- Missing data removal: In this process, the null values such as missing values and Nan values are replaced by 0.
- Missing and duplicate values were removed and data was cleaned of any abnormalities.
- Label Encoding: In this process, the string values are converted into integer for more prediction.

Data Splitting

- During the machine learning process, data are needed so that learning can take place.
- In addition to the data required for training, test data are needed to evaluate the performance of the algorithm but here we have training and testing dataset separately.
- In our process, we have to divide as training and testing.
- Data splitting is the act of partitioning available data into two portions, usually for cross-validator purposes.
- One Portion of the data is used to develop a predictive model and the other to evaluate the model's performance.

Classifications

Random Forest Algorithm

- **Random forest** is a machine learning algorithm for fraud detection.
  It's an unsupervised learning algorithm that identifies fraud by isolating outliers in the data.
- Random Forest is based on the Decision Tree algorithm. It isolates the outliers by randomly selecting a feature from the given set of features and then randomly selecting a split value between the max and min values of that feature.
- This random partitioning of features will produce shorter paths in trees for the fraud data points, thus distinguishing them from the rest of the data.
- Random Forest isolates fraud in the data points instead of profiling non fraud data points. As fraud data points mostly have a lot shorter tree paths than the normal data points, trees in the isolation forest does not need to have a large depth so a smaller max_depth can be used resulting in low memory requirement.

KNN Algorithm:

- K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.

- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.
- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.
- K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.
- It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

7. SCREEN SHOTS

DATA SELECTION:

```
#--------------------Data Selection---------------#
*******************

    step      type      amount  ... newbalanceDest  isFraud  isFlaggedFraud
0      1   PAYMENT         NaN  ...            0.00        0               0
1      1   PAYMENT     1864.28  ...            0.00        0               0
2      1  TRANSFER         NaN  ...            0.00        1               0
3      1  CASH_OUT     181.00  ...            0.00        1               0
4      1   PAYMENT    11668.14  ...            0.00        0               0
5      1   PAYMENT     7817.71  ...            0.00        0               0
6      1   PAYMENT     7107.77  ...            0.00        0               0
7      1   PAYMENT     7861.64  ...            0.00        0               0
8      1   PAYMENT     4024.36  ...            0.00        0               0
9      1     DEBIT     5337.77  ...        40348.79        0               0
10     1     DEBIT     9644.94  ...       157982.12        0               0
11     1   PAYMENT     3099.97  ...            0.00        0               0
12     1   PAYMENT     2560.74  ...            0.00        0               0
13     1   PAYMENT    11633.76  ...            0.00        0               0
14     1   PAYMENT     4098.78  ...            0.00        0               0
15     1  CASH_OUT   229133.94  ...        51513.44        0               0
16     1   PAYMENT     1563.82  ...            0.00        0               0
17     1   PAYMENT     1157.86  ...            0.00        0               0
18     1   PAYMENT      671.64  ...            0.00        0               0
19     1  TRANSFER   215310.30  ...            0.00        0               0
```

DATA PREPROCESSING

Find Missing Values

```
#--------------------Find missing values---------------#
********************
step              0
type              0
amount            2
nameOrig          0
oldbalanceOrg     0
newbalanceOrig    0
nameDest          0
oldbalanceDest    0
newbalanceDest    0
isFraud           0
isFlaggedFraud    0
dtype: int64
```

Handling Missing values:

```
#--------------------Fill 0 from missing Values---------------#
********************
step              0
type              0
amount            0
nameOrig          0
oldbalanceOrg     0
newbalanceOrig    0
nameDest          0
oldbalanceDest    0
newbalanceDest    0
isFraud           0
isFlaggedFraud    0
dtype: int64
```

Label Encoding:

```
#--------------------Before Label Encoding--------------#
*******************
    step      type     amount  ... newbalanceDest  isFraud  isFlaggedFraud
0      1   PAYMENT       0.00  ...            0.00        0               0
1      1   PAYMENT    1864.28  ...            0.00        0               0
2      1  TRANSFER       0.00  ...            0.00        1               0
3      1  CASH_OUT     181.00  ...            0.00        1               0
4      1   PAYMENT   11668.14  ...            0.00        0               0
5      1   PAYMENT    7817.71  ...            0.00        0               0
6      1   PAYMENT    7107.77  ...            0.00        0               0
7      1   PAYMENT    7861.64  ...            0.00        0               0
8      1   PAYMENT    4024.36  ...            0.00        0               0
9      1     DEBIT    5337.77  ...        40348.79        0               0
10     1     DEBIT    9644.94  ...       157982.12        0               0
11     1   PAYMENT    3099.97  ...            0.00        0               0
12     1   PAYMENT    2560.74  ...            0.00        0               0
13     1   PAYMENT   11633.76  ...            0.00        0               0
14     1   PAYMENT    4098.78  ...            0.00        0               0
15     1  CASH_OUT  229133.94  ...        51513.44        0               0
16     1   PAYMENT    1563.82  ...            0.00        0               0
17     1   PAYMENT    1157.86  ...            0.00        0               0
18     1   PAYMENT     671.64  ...            0.00        0               0
19     1  TRANSFER  215310.30  ...            0.00        0               0
```

```
#--------------------After Label Encoding--------------#
*******************
    step  type     amount  ...  newbalanceDest  isFraud  isFlaggedFraud
0      1     3       0.00  ...            0.00        0               0
1      1     3    1864.28  ...            0.00        0               0
2      1     4       0.00  ...            0.00        1               0
3      1     1     181.00  ...            0.00        1               0
4      1     3   11668.14  ...            0.00        0               0
5      1     3    7817.71  ...            0.00        0               0
6      1     3    7107.77  ...            0.00        0               0
7      1     3    7861.64  ...            0.00        0               0
8      1     3    4024.36  ...            0.00        0               0
9      1     2    5337.77  ...        40348.79        0               0
10     1     2    9644.94  ...       157982.12        0               0
11     1     3    3099.97  ...            0.00        0               0
12     1     3    2560.74  ...            0.00        0               0
13     1     3   11633.76  ...            0.00        0               0
14     1     3    4098.78  ...            0.00        0               0
15     1     1  229133.94  ...        51513.44        0               0
16     1     3    1563.82  ...            0.00        0               0
17     1     3    1157.86  ...            0.00        0               0
18     1     3     671.64  ...            0.00        0               0
19     1     4  215310.30  ...            0.00        0               0
```

DATA SPLITTING:

```
#--------------------Data Splitting---------------#
*******************
Total no of dataset : (80000, 11)
Training set Without Target (64000, 10)
Training set only Target (64000,)
Testing set Without Target (16000, 10)
Testing set only Target (16000,)
```

CLASSIFICATION:

```
#--------------------Random Forest Algorithm---------------#
*******************
Matrix:
[[15976     0]
 [   12    12]]
classfication:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     15976
           1       1.00      0.50      0.67        24

   micro avg       1.00      1.00      1.00     16000
   macro avg       1.00      0.75      0.83     16000
weighted avg       1.00      1.00      1.00     16000

Accuracy:  99.925
```

```
#--------------------KNN Algorithm---------------#
*******************

Matrix:
[[15975     1]
 [   24     0]]
classfication:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     15976
           1       0.00      0.00      0.00        24

   micro avg       1.00      1.00      1.00     16000
   macro avg       0.50      0.50      0.50     16000
weighted avg       1.00      1.00      1.00     16000

Accuracy:  99.84375
```

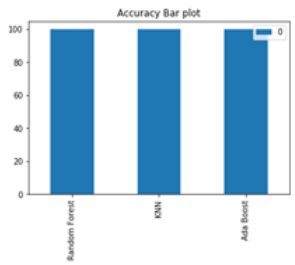
```
#--------------------Ada Boost---------------#
*******************
0.999

Matrix:
[[15973     3]
 [   13    11]]
classfication:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     15976
           1       0.79      0.46      0.58        24

   micro avg       1.00      1.00      1.00     16000
   macro avg       0.89      0.73      0.79     16000
weighted avg       1.00      1.00      1.00     16000

Accuracy:  99.9
```

PREDICTION:

```
#--------------------Get input from user---------------#
*******************

Enter the Step: 1

Enter the Type: 4

Enter the Amount: 0

Enter the nameOrig: 15121

Enter the oldbalance: 181

Enter the newbalance: 0

Enter the nameDest: 7874

Enter the oldbalance: 0

Enter the newbalance: 0

Enter the isFlaggedFraud: 0
[1]
This is financial Fraud
```

GRAPH:

```
#--------------------Camparision batween 3 Algorithm Accuracy--------------#
*******************
```



8. CONCLUSION AND FEATURE ENHANCEMENT

In this project, we propose an approach to utilise the Random Forest algorithm, KNN and Adaboost algorithm for fraud detection in financial statements. We call the approach the three algorithms on datasets with significantly reduced dimensionality. The Classifications classifier gives high accuracy results that are comparable or superior to other fraud detection techniques in spite of working with reduced data and also compared with graph

FUTURE ENHANCEMENT

In future, discovery of additional information based on cause-event Fraud detection well as prediction of detection based on cause events, etc. The working of the proposed approach in a web application.

REFERENCES

1. Albizri, D. Appelbaum, and N. Rizzotto, ‘‘Evaluation of financial statements fraud detection research: A multi-disciplinary analysis,’’ Int. J. Discl. Governance, vol. 16, no. 4, pp. 206–241, Dec. 2019.
2. R.Albright, ‘‘Taming text with the SVD.SAS institute white paper, ’’SAS Inst., Cary, NC, USA, White Paper 10.1.1.395.4666, 2004.
3. M. S. Beasley, ‘‘An empirical analysis of the relation between the board of director composition and financial statement fraud,’’ Accounting Rev., vol. 71, pp. 443–465, Oct. 1996.
4. T. B. Bell and J. V. Carcello, ‘‘A decision aid for assessing the likelihood of fraudulent financial reporting,’’ Auditing A, J. Pract. Theory, vol. 19, no. 1, pp. 169–184, Mar. 2000.
5. M.D.BeneishandC.Nichols,‘‘The predictable cost of earnings manipulation,’’Dept.Accounting,Kelley SchoolBus.,IndianaUniv.,Bloomington, IN, USA, Tech. Rep. 1006840, 2007.
6. R. J. Bolton and D. J. Hand, ‘‘Statistical fraud detection: A review,’’ Stat. Sci., vol. 17, no. 3, pp. 235–249, Aug. 2002.

7. M.Cecchini, H.Aytug, G.J.Koehler, and P.Pathak,‘‘Making words work: Using financial text as a predictor of financial events,’’ Decis. Support Syst., vol. 50, no. 1, pp. 164–175, 2010.
8. Q. Deng, ‘‘Detection of fraudulent financial statements based on naïve Bayes classifier,’’ in Proc. 5th Int. Conf. Comput. Sci. Educ., 2010, pp. 1032–1035.
9. S. Chen, Y.-J.-J. Goo, and Z.-D. Shen, ‘‘A hybrid approach of stepwise regression, logistic regression, support vector machine, and decision tree for forecasting fraudulent financial statements,’’ Sci. World J., vol. 2014, pp. 1–9, Aug. 2014.
10. X. Chen and R. Ye, ‘‘Identification model of logistic regression analysis on listed Firms’ frauds in China,’’ in Proc. 2nd Int. Workshop Knowl. Discovery Data Mining, Jan. 2009, pp. 385–388.
11. Chimonaki, S. Papadakis, K. Vergos, and A. Shahgholian, ‘‘Identification of financial statement fraud in greece by using computational intelligencetechniques,’’inProc.Int.WorkshopEnterpriseAppl.,Markets Services Finance Ind. Cham, Switzerland: Springer, 2018, pp. 39–51.
12. R. Cressey, ‘‘Other people’s money; a study of the social psychology of embezzlement,’’ Amer. J. Sociol., vol. 59, no. 6, May 1954, doi: 10.1086/221475.
13. B. Dbouk and I. Zaarour, ‘‘Towards a machine learning approach for earningsmanipulationdetection,’’AsianJ. Bus.Accounting,vol.10,no.2, pp. 215–251, 2017.
14. Q. Deng, ‘‘Application of support vector machine in the detection of fraudulent financial statements, ’’inProc.4thInt.Conf.Comput.Sci.Educ., Jul. 2009, pp. 1056–1059.
15. S. Chen, ‘‘Detection of fraudulent financial statements using the hybrid data mining approach,’’ SpringerPlus, vol. 5, no. 1, p. 89.