FRAUD DETECTION ON BANK TRANSACTION

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

MASTER OF COMPUTER APPLICATIONS

Submitted By

RAJANA LAKSHMI
Reg No: 2251926027

Under the esteemed guidance of

Mr. GODDU RAMAKRISHNA, M. Tech ,ph.D
Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COLLEGE OF ENGINEERING

Dr. B.R. AMBEDKAR UNIVERSITY, SRIKAKULAM

2023-2024

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
COLLEGE OF ENGINEERING
Dr. B.R. AMBEDKAR UNIVERSITY, SRIKAKULAM

CERTIFICATE

This is to certify that the project entitled "**FRAUD DETECTION ON BANK TRANSACTION**" that is being submitted by **RAJANA LAKSHM 2251926027** in partial fulfilment of requirements for the award of the degree in **MASTER OF COMPUTER APPLICATIONS** during **2023 - 2024**, in **Dr. B. R. AMBEDKAR UNIVERSITY, SRIKAKULAM, COLLEGE OF ENGINEERING** is a record of bonafide work carried out by him under our guidance and super vision. The results embodied in this work have not being submitted to any other university or institute for the award of any degree or diploma.

SUPERVISOR

Dr. G . RAMAKRISHNA,M.Tech..,Ph.D.
Assistant professor

HEAD OF THE DEPARTMENT

Mr. .R. SRIDHAR M.Tech
Assistant professor

External examiner

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
COLLEGE OF ENGINEERING
Dr. B.R. AMBEDKAR UNIVERSITY, SRIKAKULAM

DECLARATION

I hereby declare that the project work entitled **“FRAUD DETECTION ON BANK TRANSACTION”** submit by me for the award of the degree of **MASTER OF COMPUTER APPLICATIONS**, under the guidance of **DR.G.RAMAKRISHNA M.Tech,ph..D** Assistant Professor. **Dr. B.R. AMBEDKAR UNIVERSITY SRIKAKULAM.** Is original and it has not been submitted earlier.

Signature of candidate

RAJANA LAKSH MI

Regd.No:2251926027

Place: Srikakulam

Date:

AKNOWLEDGEMENT

A Project is a golden opportunity for learning and self-development. I consider myself lucky and privileged to express my gratefulness and deep sense of gratitude and to our guide **Dr. G. RAMAKRISHNA M.Tech. Ph.D** Assistant professor, Department of **COMPUTER SCIENCE AND ENGINEERING**, for stimulating suggestions and encouragement that helped me at every stage of our project work, which made this project successful.

I also express my gratefulness gratitude to our **Mr . R.SRIDHAR M.Tech.**, Head of the Department of Computer Science and Engineering , **Dr.B.R. AMBEDKAR UNIVERSITY, SRIKAKULAM** for his support and encouragement throughout the project.

I also express my gratefulness gratitude to our **Prof. Dr.Ch.RAJSEKHARA RAO M.Tech. Ph,D., Principal,** College of engineering**,** **Dr.B.R.AMBEDKAR UNIVERSITY, SRIKAKULAM** for his support and encouragement throughout the project.

Further-more, I would also like to acknowledge my thankfulness with much appreciation the crucial role of our TEACHING STAFF, NONTEACHING STAFF, PARENTS AND FRIENDS for their love, support, encouragement and cooperation.

Place:Srikakulam RAJANA LAKSHMI
Date: Reg No:225192602

ABSTRACT

The banking sector is a very important sector in our present day generation where almost every human has to deal with the bank either physically or online. In dealing with the banks, the customers and the banks face the chances of been trapped by fraudsters. Examples of fraud include insurance fraud, credit card fraud, accounting fraud, etc. Detection of fraudulent activity is thus critical to control these costs. The most common types of bank fraud include debit and credit card fraud, account fraud, insurance fraud, money laundering fraud, etc. Bankers are obliged to safeguard their financial assets as well as institutional integrity to armored the global financial system. Anti-fraud guard systems are regularly circumvented by fraudsters' dodging techniques. This paper hereby addresses bank fraud detection via the use of machine learning techniques; association, clustering, forecasting, and classification to analyze the customer data in order to identify the patterns that can lead to frauds. Upon identification of the patterns, adding a higher level of verification/authentication to banking processes can be added

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER-1

INTRODUCTION

CHAPTER-1

1. INTRODUCTION

According to The American Heritage dictionary, second college edition, fraud is defined as a deception deliberately practiced to secure unfair unlawful gain. Fraud detection is the recognition of symptoms of fraud where no prior suspicion or tendency to fraud exists. Examples include insurance fraud, credit card fraud and accounting fraud. Data from the Nigeria Inter-Bank Settlement System (NIBSS) has revealed that fraudulent transactions in the banking sector at its peak. Fraud has evolved from being committed by casual fraudsters to being committed by organized crime and fraud rings that use sophisticated methods to take over control of accounts and commit fraud. Some 6.8 million Americans were victimized by bank transaction fraud in 2007, according to Javelin research. Such fraud on existing accounts accounted for more than $3 billion in losses in 2007. The Nilson Report estimates the cost to the industry to be $4.84 billion. Javelin estimates the losses at more than six times that amount – some $30.6 billion in 2007. Of course, fraud is not a domestic product as it‘s everywhere. For instance, card fraud losses cost UK economy GBP 423 million in 2006. Bank transaction fraud accounts for the biggest cut of the $600 million that airlines lose each year globally.1.01

1.1 OVERVIEW

Fraud detection is a set of activities undertaken to prevent money or property from being obtained through false pretenses. Fraud detection is applied to many industries such as banking or insurance.

In banking, fraud may include forging checks or using stolen credit cards. With an unlimited and rising number of ways someone can commit fraud, detection can be difficult. Activities such as reorganization, downsizing, moving to new information systems or encountering a cyber security breach could weaken an organization's ability to detect fraud. Techniques such as real-time monitoring for fraud are recommended. Organizations should look for fraud in financial transactions, locations, devices used, initiated sessions and authentication systems.

Fraud can be committed in different ways and different settings. For example, fraud can be committed in banking, insurance, government and healthcare sectors. A common type of banking fraud is customer account takeover. This is when someone illegally gains access to a victim's bank account using bots. Other examples of fraud in banking include the use of malicious applications, the use of false identities, money laundering, credit card fraud and mobile fraud.

Government fraud is committing fraud against federal agencies such as the U.S. Department of Health and Human Services, Department of Transportation, Department of Education or Department of Energy. Types of government fraud include billing for unnecessary procedures, overcharging for items that cost less, providing old equipment when billing for new equipment and reporting hours worked for a worker that does not exist.

1.2 MACHINE LEARNING

Machine learning could be a subfield of computer science (AI). The goal of machine learning typically is to know the structure information of knowledge of information and match that data into models which will be understood and used by folks. Although machine learning could be a field inside technology, it differs from ancient process approaches.

In ancient computing, algorithms are sets of expressly programmed directions employed by computers to calculate or downside solve. Machine learning algorithms instead give computers to coach on knowledge inputs and use applied math analysis so as to output values that fall inside a particular vary. thanks to this, machine learning facilitates computers in building models from sample knowledge to modify decision-making processes supported knowledge inputs.

1.3 MACHINE LEARNING STRATEGIES

In machine learning, tasks square measure typically classified into broad classes. These classes square measure supported however learning is received or however feedback on the educational is given to the system developed. Two of the foremost wide adopted machine learning strategies square measure supervised learning that trains algorithms supported example input and output information that's tagged by humans, and unattended learning that provides the algorithmic program with no tagged information so as to permit it to search out structure at intervals its computer file.

1.3.1 SUPERVISED LEARNING

In supervised learning, the pc is given example inputs that square measure labelled with their desired outputs. The aim of this technique is for the algorithmic program to be ready to ―learn‖ by comparison its actual output with the ―taught‖ outputs to search out errors, and modify the model consequently. Supervised learning thus uses patterns to predict label values on extra unlabeled information. For example, with supervised learning, an algorithm may be fed data with images of sharks labelled as fish and images of oceans labelled as water. By being trained on this data, the supervised learning algorithm should be able to later identify unlabeled shark images as fish and unlabeled ocean images as water.

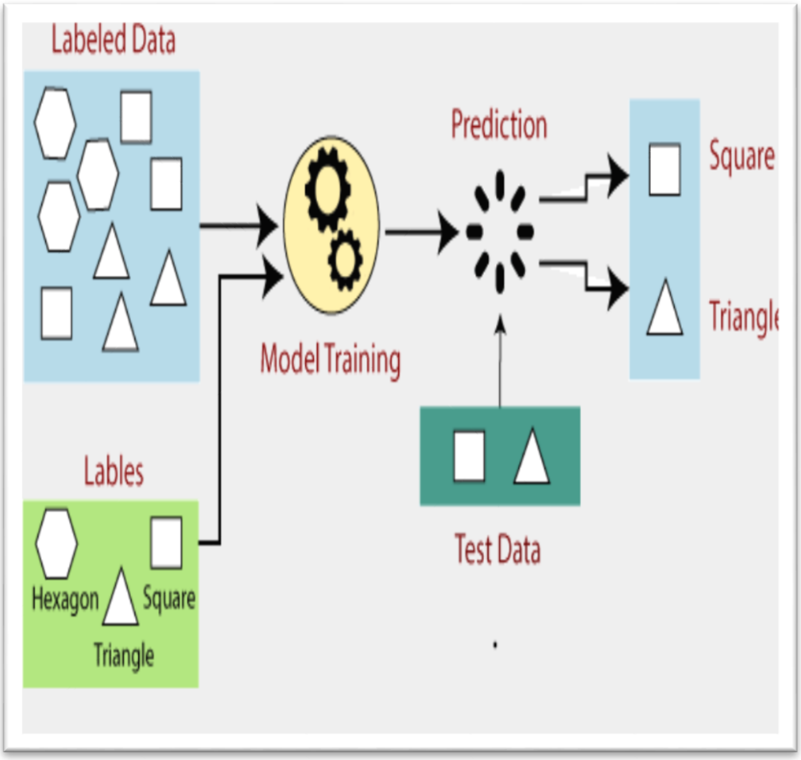

Fig1.1 SUPERVISED LEARNING

A common use case of supervised learning is to use historical information to predict statistically probably future events. It's going to use historical stock exchange info to anticipate approaching fluctuations or be used to filter spam emails. In supervised learning, labeled photos of dogs are often used as input file to classify unlabeled photos of dogs.

1.3.2 UNSUPERVISED LEARNING

In unsupervised learning, information is unlabeled, that the learning rule is left to seek out commonalities among its input file. The goal of unsupervised learning is also as easy as discovering hidden patterns at intervals a dataset;however it should even have a goal of feature learning, that permits the procedure machine to mechanically discover the representations that square measure required to classify data.

Unsupervised learning is usually used for transactional information. You will have an oversized dataset of consumers and their purchases, however as a person's you'll probably not be able to add up of what similar attributes will be drawn from client profiles and their styles of purchases.

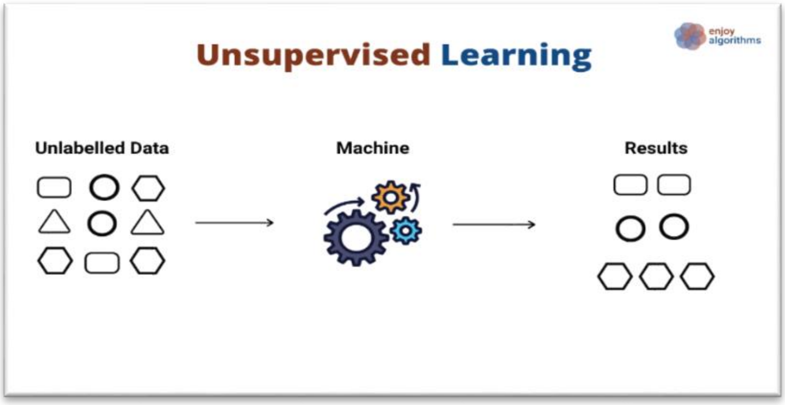

Fig 1.2 UNSUPERVISED LEARNING

With this information fed into Associate in Nursing unattended learning rule, it should be determined that ladies of a definite age vary UN agency obtain unscented soaps square measure probably to be pregnant, and so a promoting campaignassociated with physiological condition and baby will be merchandised.

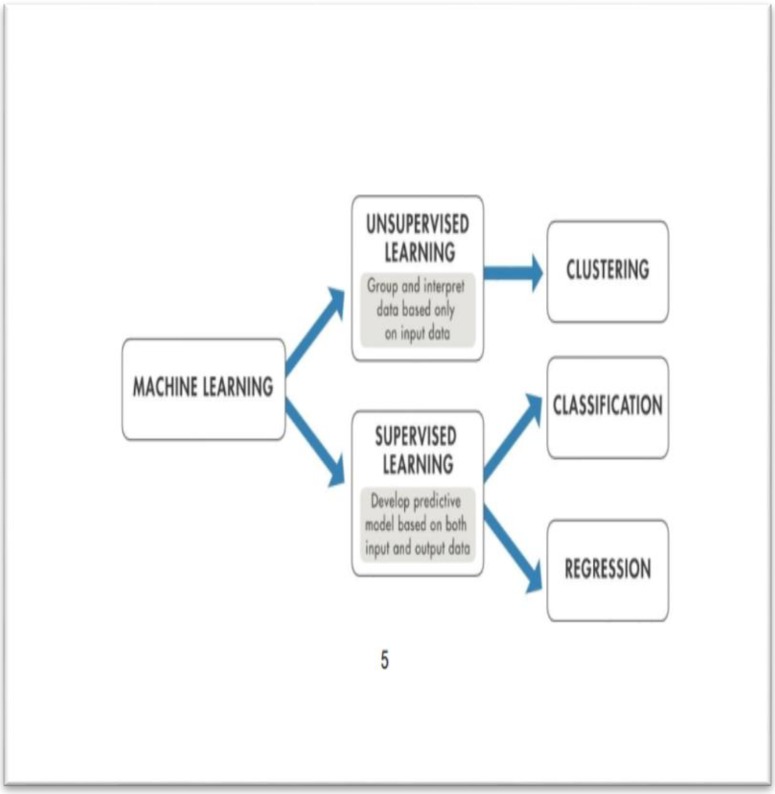
MACHINE LEARNING
UNSUPERVISED LEARNING
Group and interpret data based only on input data
SUPERVISED LEARNING
Develop predictive model based on both input and output data
CLUSTERING
CLASSIFICATION
REGRESSION
5

Fig1.3 MACHINE LEARNING

1.4 MACHINE LEARNING CLASSIFICATION

Classification is a supervised machine learning method where the model tries to predict the correct label of a given input data. In classification, the model is fully trained using the training data, and then it is evaluated on test data before being used to perform prediction on new unseen data.

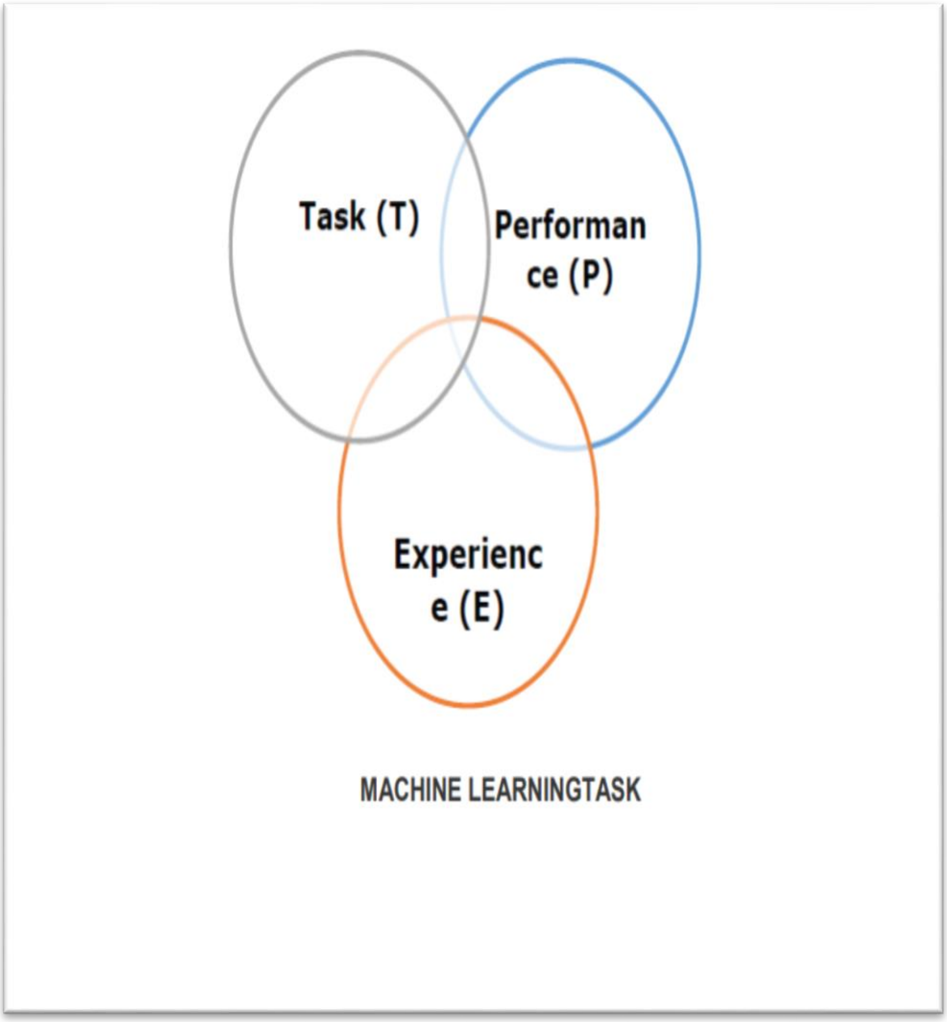

Fig1.4

1.5 MACHINE LEARNING REGRESSION

Regression is a supervised machine learning technique which is used to predict continuous values. The ultimate goal of the regression algorithm is to plot a best-fit line or a curve between the data. The three main metrics that are used for evaluating the trained regression model are variance, bias and error.

1.6 MACHINE LEARNING CLUSTERING

Clustering is a data science technique in machine learning that groups similar rows in a data set. After running a clustering technique, a new column appears in the data set to indicate the group each row of data fits into best.

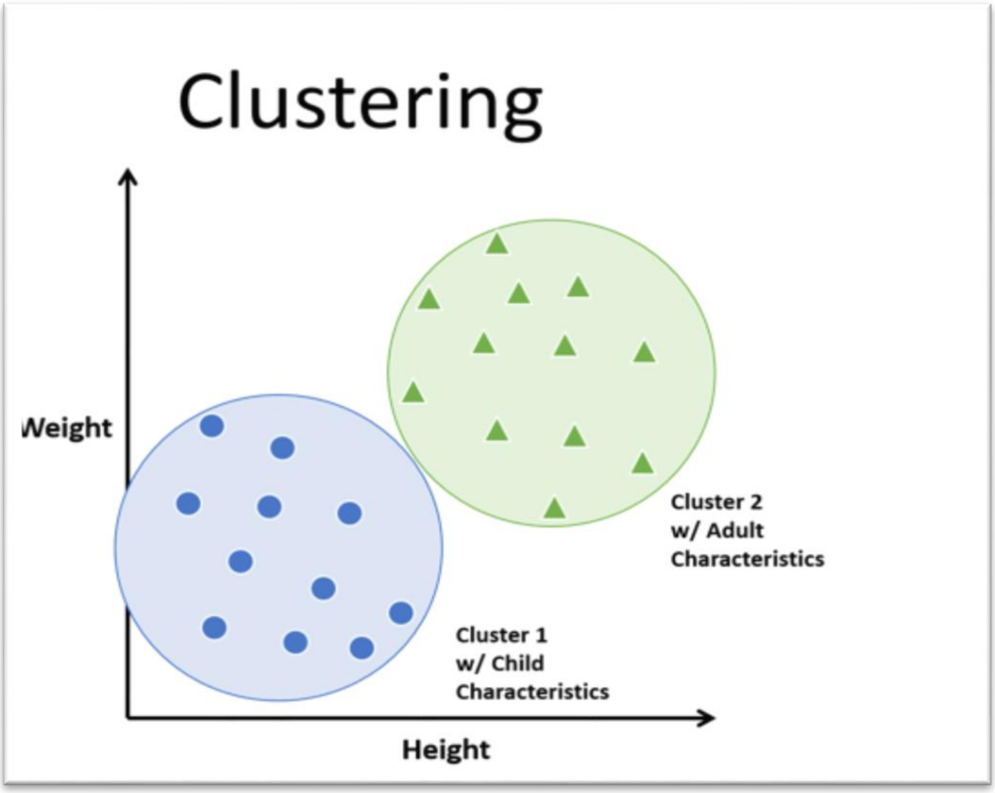

Fig 1.5

CHAPTER-2

LITERATURE SURVEY

CHAPTER 2

2. LITERATURE SURVEY

Fraud detection has been usually seen as a data mining problem where the objective is to correctly classify the transactions as legitimate or fraudulent. For classification problems many performance measures are defined most of which are related with correct number of cases classified correctly.

A more appropriate measure is needed due to the inherent structure of credit card transactions. When a card is copied or stolen or lost and captured by fraudsters it is usually used until its available limit is depleted. Thus, rather than the number of correctly classified transactions, a solution which minimizes the total available limit on cards subject to fraud is more prominent.

Since the fraud detection problem has mostly been defined as a classification problem, in addition to some statistical approaches many data mining algorithms have been proposed to solve it. Among these, decision trees and artificial neural networks are the most popular ones. The study of Bolton and Hand provides a good summary of literature on fraud detection problems.

However, when the problem is approached as a classification problem with variable misclassification costs as discussed above, the classical data mining algorithms are not directly applicable; either some modifications should be made on them or new algorithms developed specifically for this purpose are needed. An alternative approach could be trying to make use of general purpose meta heuristic approaches like genetic algorithm

2.1 AIM OF PROJECT

The primary aim of project's to show how to detect the fraud in the banking using some security questions given by the user as a inputs. Fraud detection is a challenging task for many banks and online payment companies. In today's life, hackers are more convenient to get personal information and powerful password decoding tools to do online fraud. Online transaction fraud causes a loss around billions of dollars for customers per year .Hence, many financial and online transation companies devoted many dollars to develop fraud detection algorithms. Data mining, data warehouse, big data and machine learning are need.

2.2 OBJECTIVES

The main aim of fraud detection systems is to detect fraud accurately and before fraud is committed. The goal is to detect and accurate false fraud detection. There are several methodologies for detecting credit card fraud, like neural networks, Genetic Algorithms, k-means clustering. fraud has been a major issue in sectors like banking, medical, insurance, and many others. Due to the increase in online transactions through different payment options, such as credit/debit cards, PhonePe, Gpay, Paytm, etc., fraudulent activities have also increased. Moreover, fraudsters or criminals have become very skilled in finding escapes so that they can loot more. Since no system is perfect and there is always a loophole them, it has become a challenging task to make a secure system for authentication and preventing customers from fraud. . So, Fraud detection algorithms are very useful for preventing frauds. Large amounts of such tagged data are fed into the supervised learning model in order to train it in such a way that it gives a valid output.

2.1 TYPES OF FRAUDS:

The types of retail fraud:

1. **Transaction fraud:**

   It is also called card-not-present (CNP) fraud where the fraudster uses a stolen credit card for online purchases. The company loses money when the original owner of the card demands a chargeback

2. **Return fraud:**

   As the e-commerce industry eased its return policies for the convenience of its customers, these became the favoured target for fraudsters to exploit and abuse. Some of the most common instances are wardrobing, receipt fraud, price switching or open box fraud, price arbitrage, and bricking

3. **Chargeback guarantee fraud:**

   Many online retail fraud prevention solutions guarantee that they will block all transactions and friendly frauds and even pay the admin fee out of their pocket

4. **Triangulation Fraud:**

   Triangulation fraud is when a customer makes a genuine purchase on a third-party marketplace (for example Amazon or Sears.com), but the product they receive was fraudulently purchased from a different retailer’s website. This practice harms businesses of all kinds Customers usually aren’t aware.

2.2Techniques for the avoid Fraud in retail

1. Predictive analytics:

It leverages analytics tools and platforms for large-scale customers and transactional data to detect fraudulent activity linked to previous incidents of fraud.

AI fraud detection solutions backed by predictive analytics can synchronies with retail payment processing infrastructure at the point of sale. Then ML algorithms that fuel fraud detection systems learn to identify Trends and characteristics links.

2.Anomaly Detection:

AI fraud detection systems for retail transactions function by analyzing massive amounts of previous and contemporary transaction data to discover underlying motives and detect anomalies.

When an anomaly is spotted, AI-driven anomaly analytics solutions can restrict a user, a transaction, or inform retailers, depending on the documented principles.

Benefits:

Quick and accurate detection of potential frauds. Reduces the cost incurred due to fraudulent activities. Impactful real-time data processing.

CHAPTER-3

METHODOLOGY

CHAPTER 3
METHODOLOGY

3.1 EXISTING SYSTEM

- In case of bank fraud detection, the existing system is detecting the fraud after fraud has been happen. Existing system maintain the large amount of data when customer comes to know about inconsistency in transaction, he/she made complaint and then fraud detection system start it working. It first tries to detect that fraud has actually occur after that it transactions that was used to fraud detection mechanism developed by master and visa cards.
- A machine learning paradigm classification, with Bank Fraud Detection being the base.
- Intrusion detections to track fraud location and so on. In case of existing system there is no confirmation of recovery of fraud and Customer satisfaction.
- Secure electronic system used to analyze the behavior of legitimate users.
- Data Mining mechanisms to classify and preprocess the user‘s data.
- Genetic algorithms.

DISADVANTAGES OF EXISTING SYSTEM

- Each payment system has its limits regarding the maximum amount in the account, the number of transactions per day and the amount of output.

- If Internet connection fails, you cannot get to your online account.

3.2 Fraud Detection Methodologies

1. **Data Collection**:

Gather transactional data from various sources such as ATM transactions, online transactions, mobile banking, etc. Include relevant data points like transaction amount, location, time, type of transaction, device used, etc.

2. **Data Prprocessing:**.

Data preprocessing is a crucial step in preparing raw transactional data for analysis and fraud detection. Here's an overview of the key steps involved in data preprocessing for bank transaction fraud detection.

- **Data cleaning:**
  Identify and handle missing values: in the dataset. This can involve techniques like imputation (replacing missing values with a calculated estimate) or removal of records with missing values if they are insignificant.
  Remove duplicates: Check for and remove duplicate records to ensure data integrity.

  Correct inaccuracies: Identify and correct any inaccuracies or inconsistencies in the data, such as typos, erroneous entries, or data entry mistakes.

- **Feature Engineering:**

Transaction aggregation: Aggregate transaction data to create new features such as total transaction amount, average transaction frequency, or total number of transactions within a specific time period for each account.

Time-based features: Extract features related to transaction timestamps, such as hour of the day, day of the week, or month of the year, which may reveal patterns in fraudulent activity.

Customer profiling: Create customer profiles based on transaction history, including features such as average transaction amount, transaction frequency, or geographical location.

- **Normalization and Scaling:**

Normalize numerical features: Scale numerical features to a common scale to prevent features with larger magnitudes from dominating the model training process. Common techniques include min-max scaling or z-score normalization.

Scale skewed distributions: Transform skewed numerical distributions using techniques such as logarithmic transformation to improve model performance.
Encoding Categorical Variables:

- **Encoding Categorical Variables:**

Convert categorical variables into numerical representations using techniques such as one-hot encoding or label encoding, enabling them to be used in machine learning models.

Handling Imbalanced Classes:

Address class imbalance if present in the dataset by employing techniques such as oversampling (e.g., SMOTE), undersampling, or using algorithms that handle imbalanced classes directly (e.g., ensemble methods like Random Forest or algorithms with class weights).

- **Feature Selection:**
  Select relevant features that are most predictive of fraudulent behavior to improve model efficiency and performance. Techniques such as feature importance from tree-based models or recursive feature elimination can help identify important features**.**
- **Data Splitting:**
  Split the preprocessed data into training, validation, and test sets for model development, evaluation, and testing, respectively. Typically, the data is split into a large portion for training and smaller portions for validation and testing.
- **Data Standardization:**
  Standardize or normalize the data to ensure consistency and comparability across different features or variables. This is particularly important for algorithms sensitive to feature scales, such as distance-based algorithms.
- **Data Transformation:**
  Apply transformations such as PCA (Principal Component Analysis) or feature scaling to reduce dimensionality or improve the interpretability of the data while preserving its essential information**.**
- **Data Privacy and Security:**
  Ensure compliance with data privacy regulations and security standards by anonymizing sensitive information, encrypting data during transmission and storage, and implementing access controls to restrict data access to authorized personnel**.**

3 .Feature Engineering:

Extract meaningful features from raw transaction data that can be used for fraud detection.Engineer new features based on transaction patterns, customer behavior, and contextual information.

4. Model Development:

Select appropriate machine learning algorithms and techniques for fraud detection, such as supervised learning, anomaly detection, or ensemble methods. Split the data into training, validation, and test set Train and evaluate multiple models using cross-validation techniques to assess performance.Tune hyperparameters and optimize model architectures for improved accuracy and generalization.

5. Model Evaluation and Validation:

Evaluate the performance of trained models using evaluation metrics relevant to fraud detection, such as accuracy, precision, recall, F1-score, and ROC-AUC.Validate models using holdout datasets or real-world testing scenarios to ensure robustness and effectiveness. Perform sensitivity analysis to understand model behavior under different thresholds and settings

.

6. Deployment and Integration:

Deploy the selected models into production systems for real-time or batch processing. Integrate the fraud detection system with banking infrastructure, including transaction processing systems, customer databases, and reporting tools. Implement APIs and interfaces for seamless data exchange and interoperability with existing systems.

7.Monitoring and Updating: Regularly monitor the model's performance and update it as necessary to adapt to new fraud patterns or changes in transaction behavior.

Throughout the project, ensure compliance with data privacy regulations and implement security measures to protect sensitive customer information. Additionally, consider integrating advanced techniques such as anomaly detection, behavior analysis, and network analysis to enhance the fraud detection system's effectiveness.

3.3 PROPOSED SYSTEM

In proposed methodology, Detection of fraudulent activity is thus critical to control these costs. This paper hereby addresses bank fraud detection via the use of machine learning techniques; association, clustering, forecasting, and classification to analyze the customer data to identify the patterns that can lead to frauds. Upon identification of the patterns, adding a higher level of verification/authentication to banking processes can be added. These kinds of frauds can be credit card fraud, insurance fraud, accounting fraud, etc. which may lead to the financial loss to the bank or the customers. Thus, detection of these kinds of frauds are very important. Fraud detection in banking sector is based on the machine learning techniques and their collective analysis from the past experiences and the probability of how the fraudsters can steal from customers and banks. Therefore, this paper addresses the analysis of data mining techniques of how to detect frauds and overcoming it in banking sector.

3.4 ADVANTAGES OF PROPOSED SYSTEM

- To eliminate real time fraud to the lowest level.
- To increase the confidence of customers in the banking system especially for online transactions.
- To discourage fraudsters (both present and intending ones)
- The proposed system significantly strengthens the security measures of the banking infrastructure, safeguarding against various types of fraudulent activities such as unauthorized access, identity theft, and transaction manipulation**.**
- The system architecture is scalable and flexible, allowing for seamless integration with existing banking systems and future expansion to accommodate growing transaction volumes and evolving business needs**.**

CHAPTER-4

SOFTWARE AND HARDWARE REQUIREMENTS

CHAPTER-4

SOFTWARE AND HARDWAREREQUIREMENTS

4.1 HARDWARE REQUIRMENT

- System : Pentium IV 2.4GHZ
- Hard Disk : 40 Gb
- Ram : 512 Mb

4.2 SOFTWARE REQUIREMENT

- Operating system : Window 10(64 bit)
- Coding Language : PYTHON
- IDE : Eclipse

4.3 Libraries:

- **Numpy**- Library of python used for arrays computation. It has so many functions. We have used this module to change 2-dimensional array into contiguous flattened array by using ravel function.

- **Pandas**- Library of python which can be used easily. It gives speed results and easily understandable. It is a library which can be used without any cost. We have used it for data analysis and to read the dataset.

- **Matplotlib**- Library of python used for visualizing the data using graphs, scatterplots and so on. Here, we have used it for datavisualizatio.

- **Sklearn** - Scikit Learn also known as sklearn is an open-source library for python programming used for implementing machine learning algorithms. It features various classification, clustering, regression machine learning algorithms. In this it is used for importing machine learning models, get accuracy, get confusion matrix.

- **Pandas Profiling**- This is library of python which can be used by anyone free of cost. It is used for data analysis. We have used this for getting the report of the dataset.

- **Seaborn**- Is a Python data visualization library based on Matplotlib, providing a high-level interface for creating attractive and informative statistical graphics.

- **Matplotlib**- is a comprehensive Python library for creating static, interactive, and animated visualizations. It provides a wide range of plotting functions and customization options, making it one of the most widely used libraries for data visualization.

  .

- Tkinter- Tkinter is library of python used often by everyone. It is a library which is used to create GUI based applications easily. It contains so many widgets like radio button, text filed and so on. We have used this for creating account registration screen, login or register screen, prediction interface which is a GUI based application .

4.4 PROGRAMMING LANGUAGE

4.4.1 About Python

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL). This tutorial gives enough understanding on Python programming language.

4.4.2 Advantages of Python

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

- Python is a MUST for students and working professionals to become a great Software Engineer especially when they are working in Web Development Domain. I will list down some of the key advantages of learning Python.
- Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- Python is Interactive − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- Python is Object-Oriented − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- Python is a Beginner's Language − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

4.4.3 Characteristics of Python

Following are important characteristics of Python Programming-

- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- It supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

4.4.4 New Approach for building window Software

The Python Framework simplifies Windows development. It provides developers with a single approach to build both desktop applications sometimes called smart client applications and Web-Based applications. It also developers to use the same tools and skills to develop software for a verity of system ranging from handled smart phones to large server installations.

4.4.5 Applications of Python

As mentioned before, Python is one of the most widely used language over the web. I'm going to list few of them here:

- **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read** − Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** − Python's source code is fairly easy-to-maintain.

- **A broad standard library** − Python's bulk of the library is very portable and cross platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- **Extendable** − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases** − Python provides interfaces to all major commercial databases.
- **GUI Programming** − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Scalable** − Python provides a better structure and support for large programs than shell scripting.

Python - GUI Programming

Python provides various options for developing graphical user interfaces (GUIs). Most important are listed below.

- Tkinter − Tkinter is the Python interface to the Tk GUI toolkit shipped with Python. We would look this option in this chapter.
- WxPython − This is an open-source Python interface for wx Windows http://wxpython.org.
- JPython − JPython is a Python port for Java which gives Python scripts seamless access to Java class libraries on the local machine.

CHAPTER-5

SYSTEM ARCHITECTURE

CHAPTER -5

SYSTEM ARCITECTURE

5.1 SYSTEM CAPABILITIES

Fraud detection and prevention system is the core of any fraud risk management strategy. Teams choose software with functionality that works best for their workflow and business needs in general.In our whitepaper, we compared machine learning- based systems with rule-based ones and described how ML-based solutions help prevent and identify fraudulent activity across several industries.For this article, we contacted specialists from NoFraud and SAS to discuss the purposes and capabilities of anti-fraud software and get their advice on the solution choice. The final section of the article contains descriptions of several solutions available on the market.

5.2 SYSTEM FUNCTIONALITIES

It is a technical specification requirement for the software products. It is the first step in the requirement analysis process which lists the requirements of particular software systems including functional, performance and security requirements. The function of the system depends mainly on the quality hardware used to run the software with given functionality.

Usability

It specifies how easy the system must be use. It is easy to ask queries in any format which is short or long, porter stemming algorithm stimulates the desired response for user.

ROBUSTNESS

It refers to a program that performs well not only under ordinary conditions but also under unusual conditions. It is the ability of the user to cope with errors for irrelevant queries during execution.

5.3 NON FUNCTIONALITY

REQUIREMENTS Portability

It is the usability of the same software in different environments. The project can be run in any operating system. Regardless of the medium used to learn probability, be it books, videos, or course material, machine learning practitioners study probability the wrong way.

Performance

These requirements determine the resources required, time interval, throughput and everything that deals with the performance of the system. In Machine Learning it is key to be able to correctly evaluate the model being produced to guarantee that the predictions are accurately describing the intended phenomenon (disease prediction, future cost estimation, etc.).

Accuracy

The result of the requesting query is very accurate and high speed of retrieving information. The degree of security provided by the system is high and effective. Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition.

Accuracy=Number of correct predictions Total number of predictions.

Maintainability

Project is simple as further updates can be easily done without affecting its stability. Maintainability basically defines that how easy it is to maintain the system. It means that how easy it is to maintain the system, analyses, change and test the application. Maintainability of this project is simple as further updates can be easily done without affecting its stability.

5.4 Decision Trees for Classification: A Machine Learning Algorithm.

Introduction

Decision Trees are a type of Supervised Machine Learning (that is you explain what the input is and what the corresponding output is in the training data) where the data is continuously split according to a certain parameter. The tree can be explained by two entities, namely decision nodes and leaves. The leaves are the decisions or the final outcomes. And the decision nodes are where the data is split.

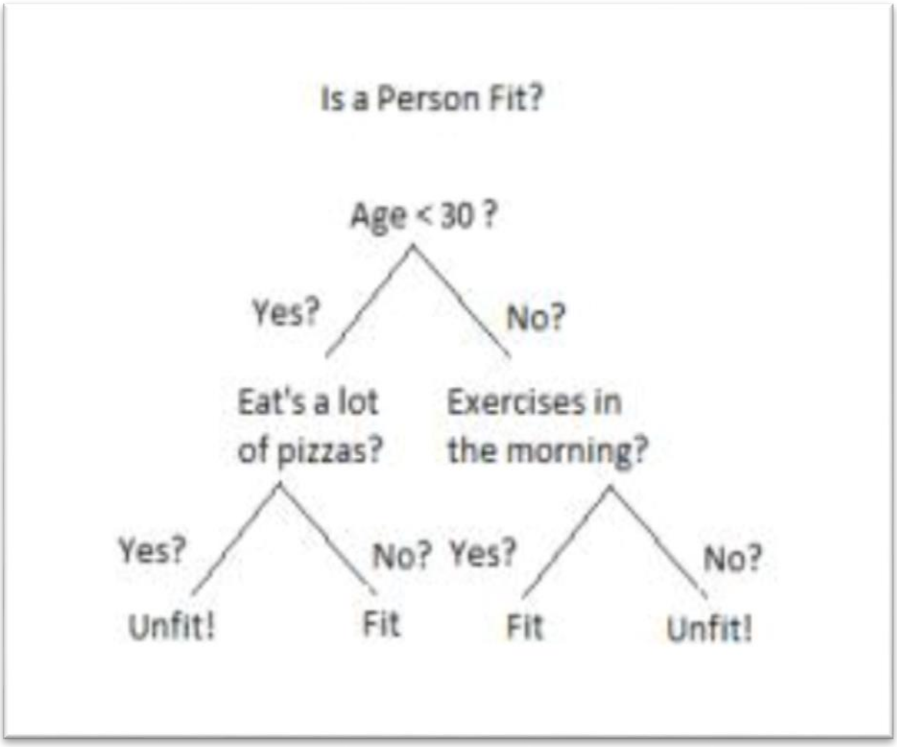

Fig 5.1

An example of a decision tree can be explained using above binary tree. Let‘s say you want to predict whether a person is fit given their information like age, eating habit, and physical activity, etc. The decision nodes here are questions like _What‘s the age?‘, _Does he exercise?‘, _Does he eat a lot of pizzas‘? And the leaves, which are

outcomes like either _fit‘, or _unfit‘. In this case this was a binary classification problem (a yes no type problem).
There are two main types of Decision Trees:

Classification trees (Yes/No types)

What we‘ve seen above is an example of classification tree, where the outcome was a variable like _fit‘ or _unfit‘. Here the decision variable is Categorical.

Here the decision or the outcome variable is Continuous.

Regression trees (Continuous data types)

Here the decision or the outcome variable is Continuous.

Working

Now that we know what a Decision Tree is, we ‘ll see how it works internally. There are many algorithms out there which construct Decision Trees, but one of the best is called as ID3 Algorithm. ID3 Stands for Iterative Dichotomiser3.

5.5 SYSTEM DESIGN AND TESTING PLAN

INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things.

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system 's relationship to help user decision-making.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

Chapter-6

SYSTEM STUDY

Chapter-6

SYSTEM STUDY

FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ♦ ECONOMICAL FEASIBILITY
- ♦ TECHNICAL FEASIBILITY
- ♦ SOCIAL FEASIBILITY

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

DATA FLOW DIAGRAM

Data Flow Diagram (DFD) is a two-dimensional diagram that describes how data is processed and transmitted in a system. The graphical depiction recognizes each source of data and how it interacts with other data sources to reach a mutual output. In order to draft a data flow diagram, one must.

- ❖ Identify external inputs and outputs.
- ❖ Determine how the inputs and outputs relate to each other

Role of DFD:

- It is a documentation support which is understood by both programmers and nonprogrammers. As DFD postulates only what processes are accomplished not how they are performed.
- A physical DFD postulates where the data flows and who processes the data.
- It permits analyst to isolate areas of interest in the organization and study them by examining the data that enter the process and viewing how they are altered when they leave.
-

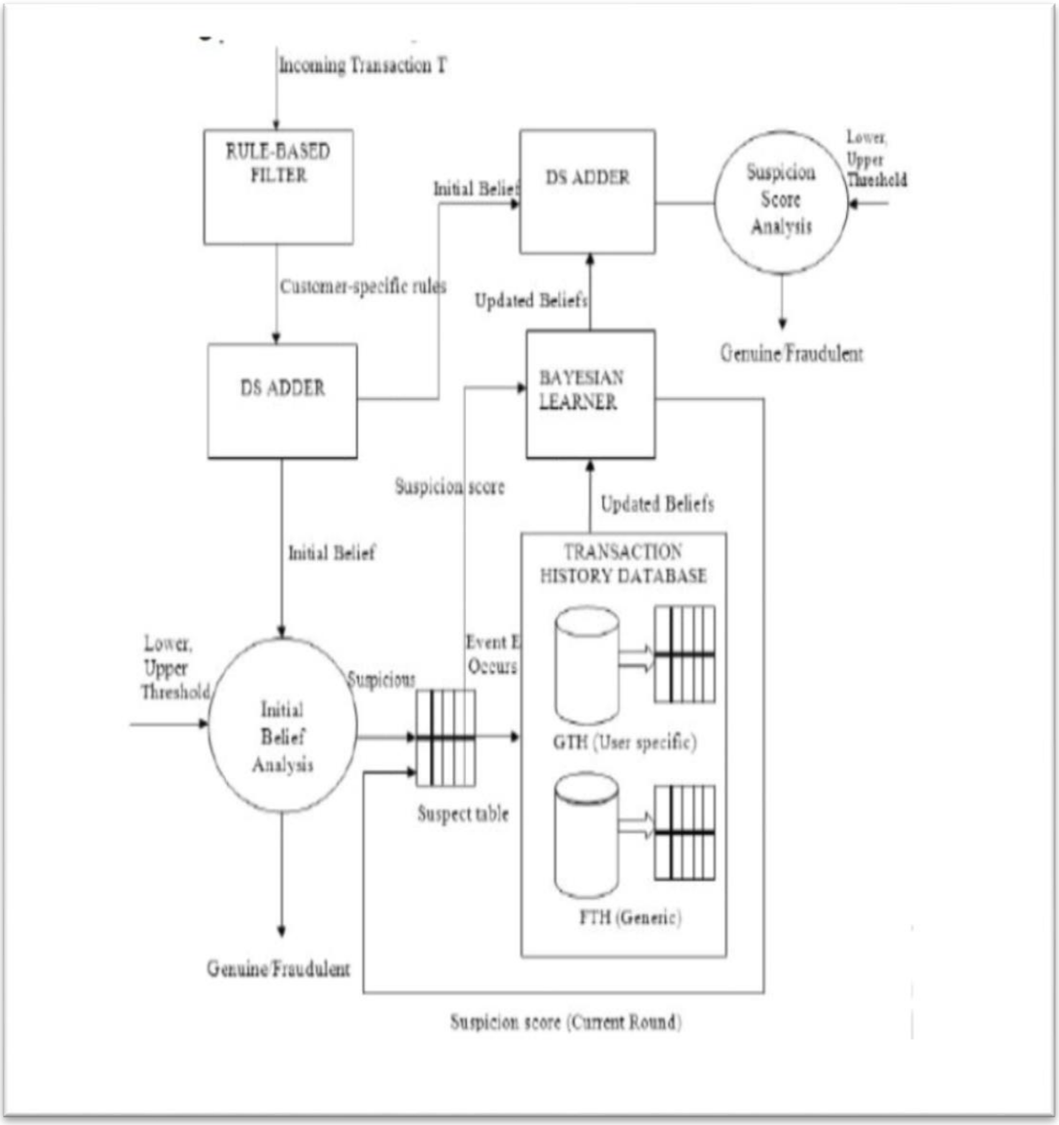

FIG 6.1

DFD-1

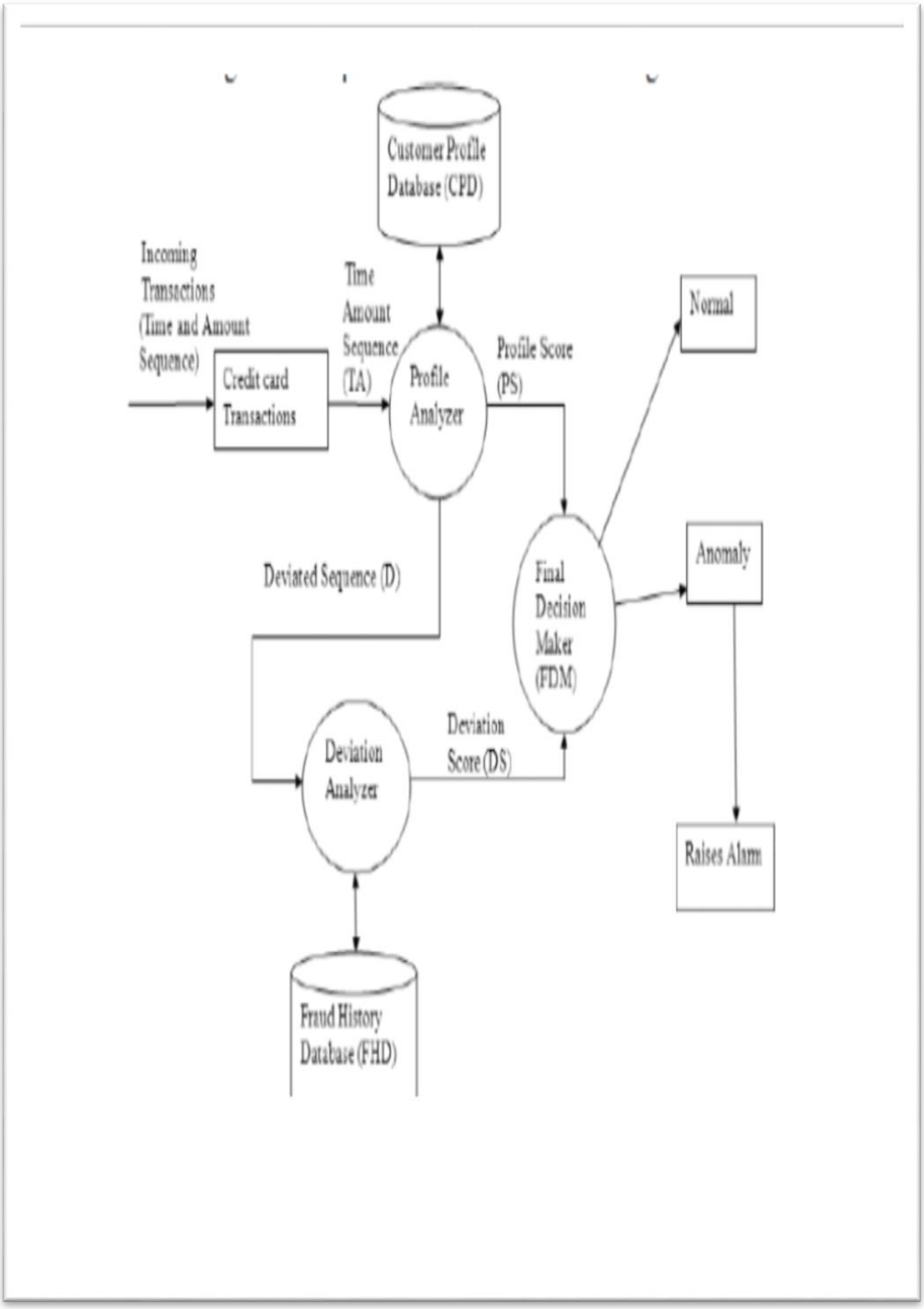
Customer Profile Database (CPD)
Incoming Transactions (Time and Amount Sequence)
Credit card Transactions
Time Amount Sequence (TA)
Profile Analyzer
Profile Score (PS)
Normal
Deviated Sequence (D)
Final Decision Maker (FDM)
Anomaly
Deviation Analyzer
Deviation Score (DS)
Raises Alarm
Fraud History Database (FHD)

FIG 6.2

DFD-2

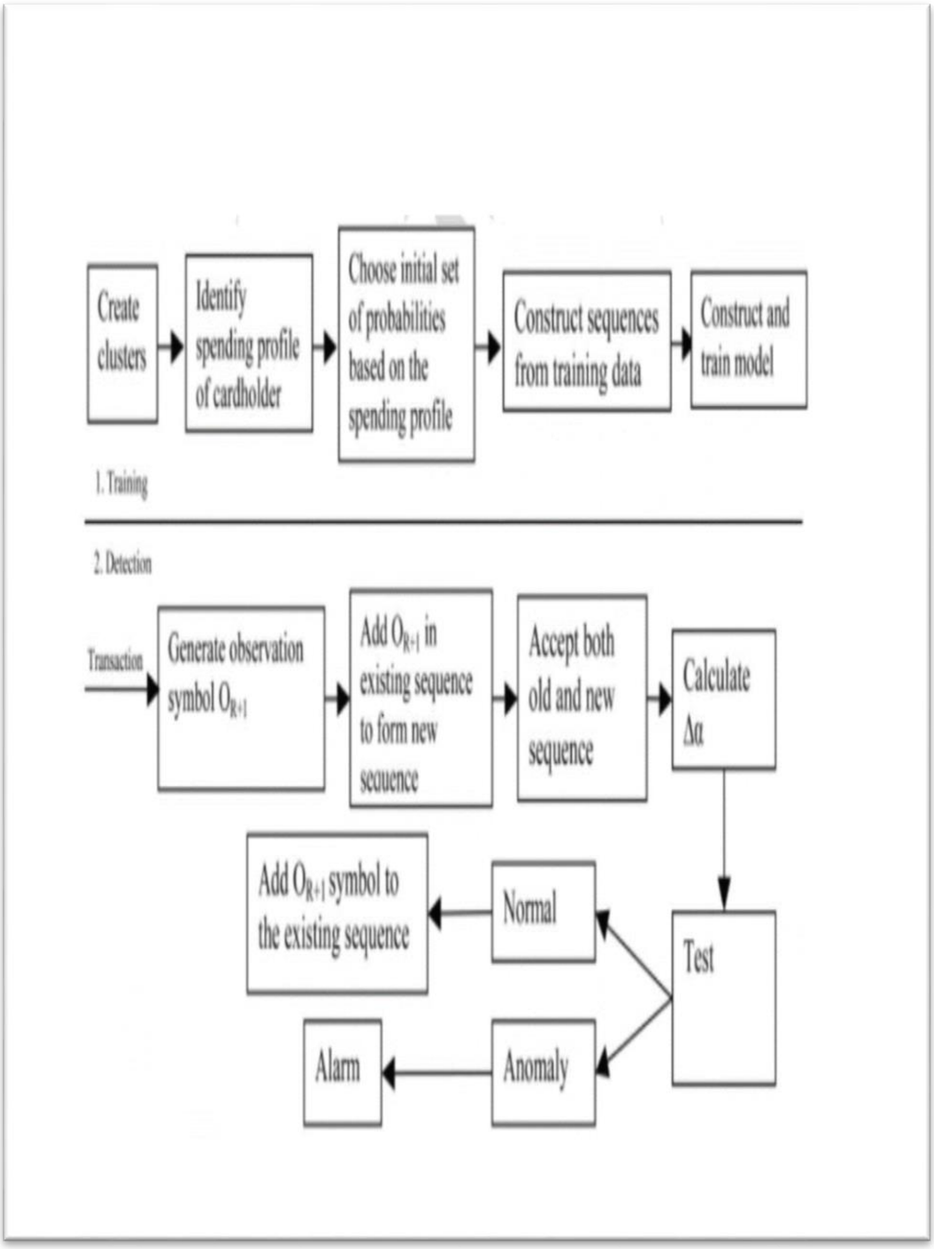
Create clusters
Identify spending profile of cardholder
Choose initial set of probabilities based on the spending profile
Construct sequences from training data
Construct and train model
1. Training
2. Detection
Transaction
Generate observation symbol O_{R+1}
Add O_{R+1} in existing sequence to form new sequence
Accept both old and new sequence
Calculate $\Delta\alpha$
Test
Normal
Anomaly
Add O_{R+1} symbol to the existing sequence
Alarm

Fig 6.3

DFD-3

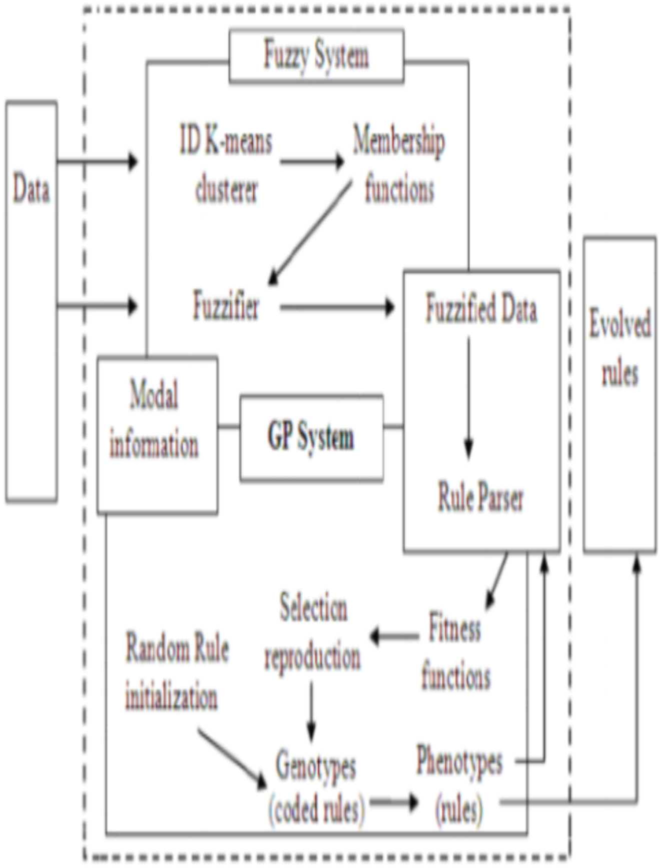
Fuzzy System
Data
ID K-means clusterer
Membership functions
Fuzzifier
Fuzzified Data
Evolved rules
Modal information
GP System
Rule Parser
Selection reproduction
Fitness functions
Random Rule initialization
Genotypes (coded rules)
Phenotypes (rules)

Fig 6.4

Chapter-7

SYSTEM TESTING

Chapter-7

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 manner. There are various types of test. Each test type addresses a specific testing requirement.

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 invasive. 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 predriven process links and integration point.

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 cannot 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 cannot ―see‖ into it. The test provides inputs and responds to outputs without considering how the software works.

7.1 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.

7.2 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.

7.3 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.

CHAPTER 8

UML DIAGRAMS

CHAPTER 8
UML DIAGRAMS

8.1 USE CASE DIAGRAM

The Use Case Diagram for the Banking System outlines the various use cases involved in using the banking system. One of the key use cases is opening an account. This involves creating a new account for a customer, which includes collecting personal information, such as name, address, and contact details.

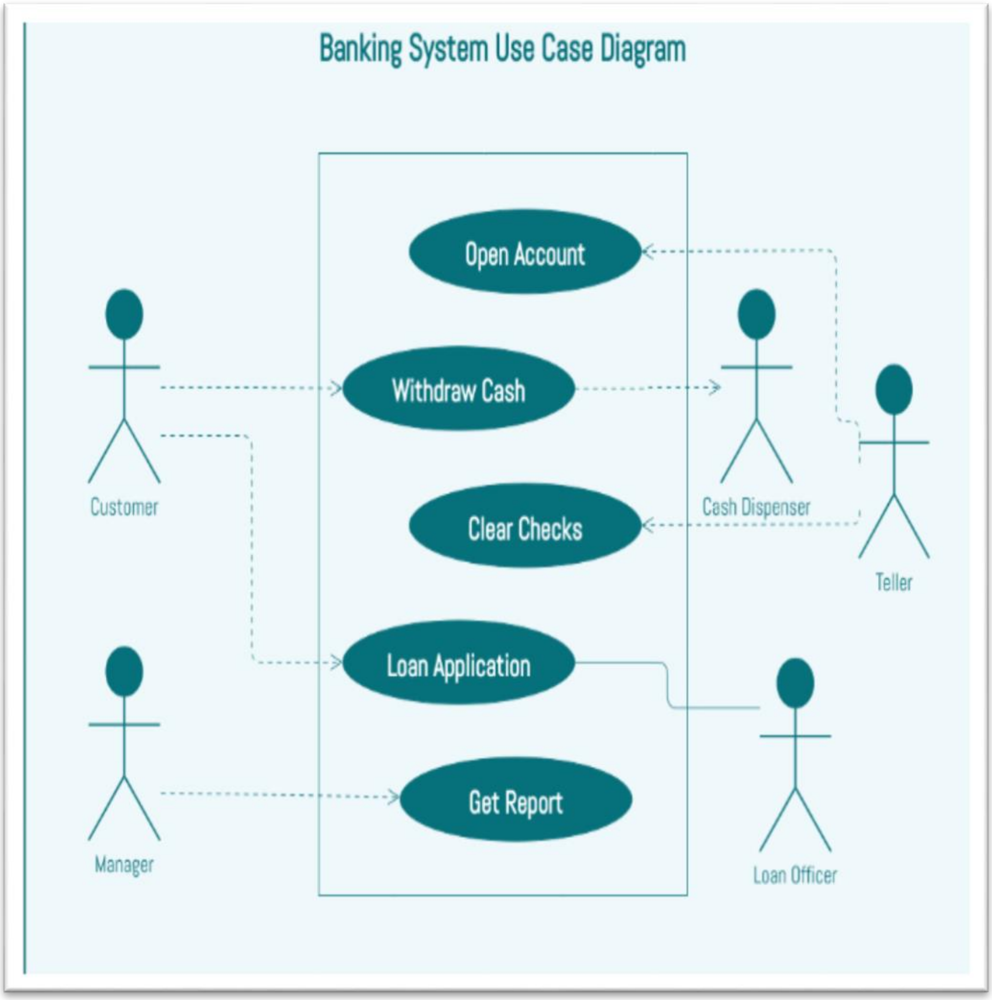

fig 8.1

8.2 FRAUD DETECTION PROCESS

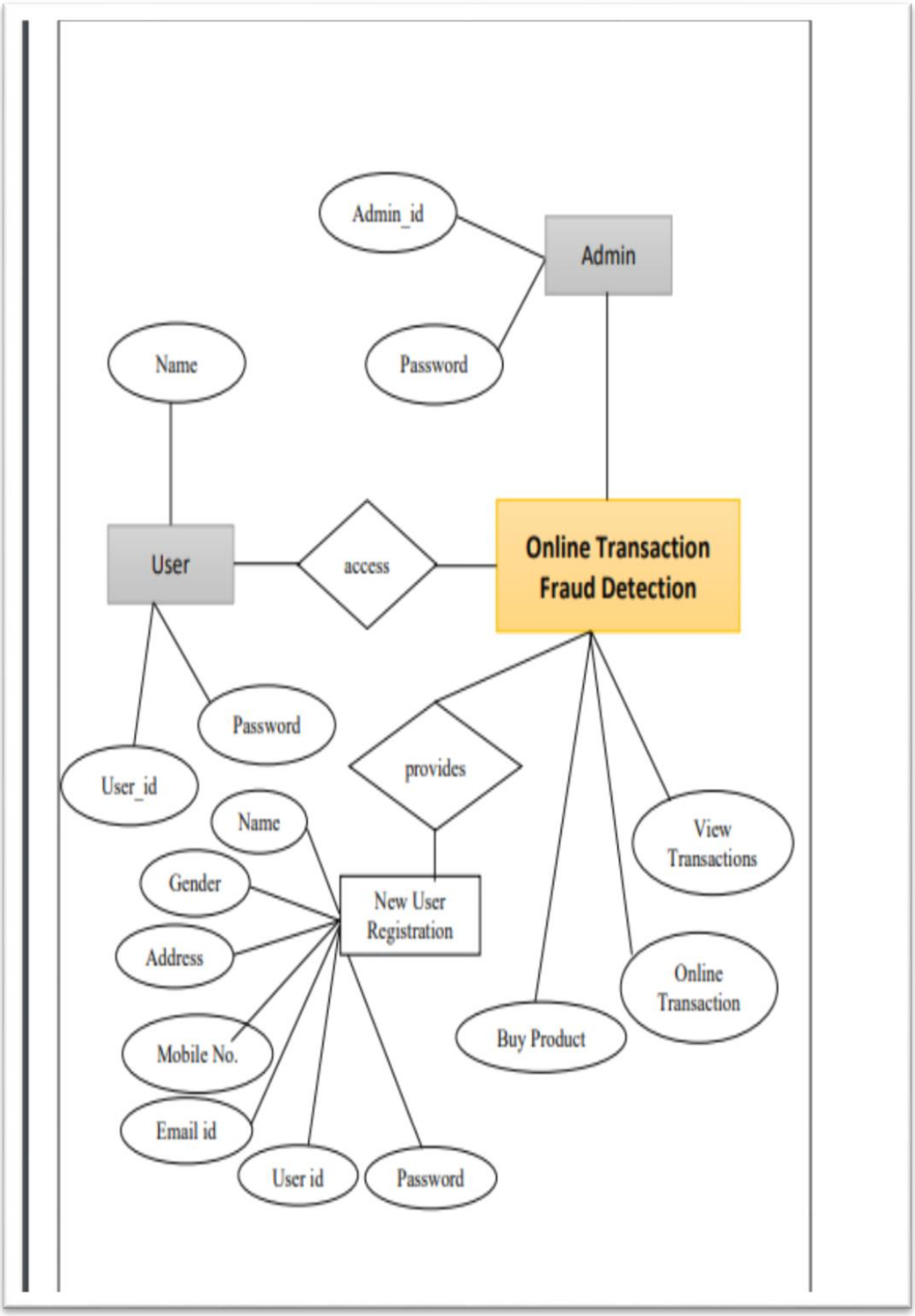

FIG 8.2

CHAPTER 9

RESULTS AND DISCUSS

CHAPTER 9
RESULTS AND DISCUSSION

Hence, we use Machine Learning for detecting fraud. Here, a machine tries to learn by itself and becomes better by experience. Also, it is an efficient way of detecting fraud because of its fast computing. It does not even require the guidance of a fraud analyst. It helps in reducing false positives for transactions as the patterns are detected by an automated system for streaming transactions that are in huge volume.

ALGORITHM	**DECISION TREE**
ACCURACY	**99%**

TABLE 8.1

CHAPTER 10

CONCLUSION

CHAPTER 10

CONCLUSION

Machine Learning is a technique used to extract vital information from existing huge amount of data and enable better decision-making for the banking and retail industries. They use data warehousing to combine various data from databases into an acceptable format so that the data can be mined. The data is then analyzed and the information that is captured is used throughout the organization to support decision-making. Data Mining techniques are very useful to the banking sector for better targeting and acquiring new customers, most valuable customer retention, automatic credit approval which is used for fraud prevention, fraud detection in real time, providing segment based products, analysis of the customers, transaction patterns over time for better retention and relation

FUTURE WORK

At Bolt, our difference is fundamental. As the first full-stack market solution to handle checkout, payments, and fraud, we have unique visibility into the full suite of checkout payment information to use for fraud detection purposes. Bolt’s competitive access to data produces ample benefits to using Bolt in terms of time, labor-hours, and saved revenue.Fraud modeling, like any other statistical modeling or machine learning, relies on having clear, consistent, normalized access to data. Very simply, without access to a particular feature (variable) that might be a strong signal, a machine-learning model is at a disadvantage. Bolt, as a payments processor and checkout flow, sees everything a merchant sees and more.Consider the types of insights Bolt might pick up that a typical fraud tool could not

REFERENCES

[1] S. M. Darwish, "An intelligent credit card fraud detection approach based on semantic fusion of two classifiers," (in English), Soft Computing, Article vol. 24, no. 2, pp. 1243-1253, Jan 2020.

[2] A. Eshghi and M. Kargari, "Introducing a new method for the fusion of fraud evidence in banking transactions with regards to uncertainty," (in English), Expert Systems with Applications, Article vol. 121, pp. 382-392, May 2019.

[3] S. Hossain, A. Abtahee, I. Kashem, M. M. Hoque, and I. H. Sarker, "Crime Prediction Using Spatio-Temporal Data," in Computing Science, Communication and Security, Singapore, 2020, pp. 277-289: Springer Singapore.

[4] M. Zamini and S. M. H. Hasheminejad, "A comprehensive survey of anomaly detection in banking, wireless sensor networks, social networks, and healthcare," (in English), Intelligent Decision Technologies-Netherlands, Article vol. 13, no. 2, pp. 229-270, 2019.

[5] I. Gonzalez-Carrasco, J. L. Jimenez-Marquez, J. L. Lopez-Cuadrado, and B. Ruiz-Mezcua, "Automatic detection of relationships between banking operations using machine learning," (in English), Information Sciences, Article vol. 485, pp. 319-346, Jun 2019.

[6] M. Pohoretskyi, D. Serhieieva, and Z. Toporetska, "The proof of the event of a financial resources fraud in the banking sector: problematic issues," (in English), Financial and Credit Activity-Problems of Theory and Practice, Article vol. 1, no. 28, pp. 36-45, 2019.

[7] K. Noor et al., "Performance analysis of a surveillance system to detect and track vehicles using Haar cascaded classifiers and optical flow method," 2017 12th IEEE Conference on Industrial Electronics and Applications (ICIE ship, risk management and marketing.).

APPENDICES

SOURCE CODE

```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

%matplotlib inline
data = pd.read_csv("OneDrive/Desktop/FRAUD DATASET.csv")
data.head()
data.info()
data.shape
data.describe()
data.isnull().sum()
data head()
data tail()
data.columns
data.type.value_counts()
labels = data['type'].astype('category').cat.categories.tolist()
counts = data['type'].value_counts()
sizes = [counts[var_cat] for var_cat in labels]
fig1, ax1 = plt.subplots()
ax1.pie(sizes,  labels=labels,  autopct='%1.1f%%',  shadow=True)
#autopct is show the % on plot
ax1.axis('equal')
plt.show()
type = data['type'].value_counts()
transactions = type.index
quantity = type.values
```

```
var = data.groupby('type').amount.sum()
fig = plt.figure()
ax1 = fig.add_subplot(1,1,1)
var.plot(kind='bar')
ax1.set_title("Total amount per transaction type")
ax1.set_xlabel('Type of Transaction')
ax1.set_ylabel('Amount');
Select only numeric columns from the DataFrame
numeric_data = data.select_dtypes(include=['number'])

# Compute the correlation matrix
correlation = numeric_data.corr()

# Print correlation with respect to the 'isFraud' column
print(correlation['isFraud'].sort_values(ascending=False))
x =
np.array(data[['type','amount','oldbalanceOrg','newbalanceOrig'
]])
y = np.array(data[['isFraud']])

from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
xtrain,xtest,ytrain,ytest = train_test_split(x,y,test_size=0.2,
random_state=42)
xtrain.shape
Assuming 'data' is your DataFrame containing features and target
x_categorical = data[['type']]  # Extract categorical feature(s)
x_numerical = data[['amount', 'oldbalanceOrg',
'newbalanceOrig']]  # Extract numerical feature(s)

# One-hot encode categorical features
encoder = OneHotEncoder()
```

```
x_categorical_encoded = encoder.fit_transform(x_categorical).toarray()

# Combine numerical and encoded categorical features
x_combined = np.concatenate((x_numerical, x_categorical_encoded), axis=1)

# Split the data into training and testing sets
xtrain, xtest, ytrain, ytest = train_test_split(x_combined, y, test_size=0.2, random_state=42)

# Initialize the DecisionTreeClassifier
model = DecisionTreeClassifier()

# Fit the model to the training data
model.fit(xtrain, ytrain)

# Evaluate the model's performance on the test data
score = model.score(xtest, ytest)
print(score)
```