Fraud Detection on Bank Payments Using Machine Learning

K RAMYA[1], B. MURALI[2]
[1]PG Student, Quba College of Engineering & Technology
[2]Assistant professor, Quba College of Engineering & Technology

Abstract*—The practice of obtaining financial gains by dishonest and unlawful means is known as financial fraud. Financial fraud, which is defined as the use of dishonest methods to obtain financial gains, has recently grown to be a serious threat to businesses and organizations. Despite several initiatives to curtail financial fraud, it continues to negatively impact society and the economy since daily losses from fraud amount to significant sums of money. Several methods for detecting fraud were first introduced many years ago. The majority of old procedures are manual, which is not only time-consuming, expensive, and inaccurate, but also unworkable. There are more studies being done, however they are ineffective at reducing losses brought on by fraud. Conventional methods for detecting these fraudulent activities, like human verifications and inspections, are inaccurate, expensive, and time-consuming. Machine-learning-based technologies can now be used intelligently to identify fraudulent transactions by examining a significant amount of financial data, thanks to the development of artificial intelligence. As a result, this study seeks to offer a novel model of fraud detection on bank payments utilizing the Random Forest Classifier Machine Learning Algorithm. Our suggested system makes use of the Banksim dataset, and we have demonstrated that it is more effective than the current system by achieving train and test accuracy of 99%.

Index Terms*— Financial Fraud, Dishonest Financial Gains, Unlawful Means, Business Threats, Economic Impact, Fraud Detection Methods, Manual Procedures, Inaccurate Methods, Time-Consuming Methods Expensive Methods

I. INTRODUCTION

With the rise of digital banking and internet transactions, banking fraud detection has become an increasingly important aspect of banking operations. Criminal activities such as identity fraud, account takeover (ATO), and credit card scams can cause significant losses for financial institutions and their customers. Traditional fraud detection methods rely on rule-based systems that can be limited in their ability to detect new and sophisticated fraud schemes. Machine learning (ML), on the other hand, offers a more advanced and accurate approach to fraud detection by analyzing vast amounts of data and identifying patterns that may indicate fraudulent behavior. Fraud detection in internet banking has also been revolutionized by the use of artificial intelligence (AI), as many businesses have incorporated this technology into their fraud analytics and systems as well. ML and AI can quickly analyze large amounts of data to detect fraudulent activities, such as unauthorized transactions or suspicious behavior patterns. With the help of AI and ML, banks can prevent financial fraud and protect their customers' assets more effectively than ever before. Businesses looking to extend the parameters of fraud detection in online banking and fintech should understand a bit about how MI can be used in this capacity—and its benefits over traditional methods..

II.LITERATURE SURVEY

1 .Building a robust mobile payment fraud detection system with adversarial examples. Authors: S. Delecourt and L. Guo. Description: Mobile payment is becoming a major payment method in many countries. However, the rate of payment fraud with mobile is higher than with credit card. One potential reason is that mobile data is easier to be modified than credit card data by fraudsters, which degrades our data-driven fraud detection system. Supervised learning methods are pervasively used in fraud detection. However, these supervised learning methods used in fraud detection have traditionally been developed following the assumption that the environment is benign; there are no adversaries trying to evade fraud detection system. In this paper, we took potential reactions of fraudsters into consideration to build a robust mobile fraud detection system using adversarial examples. Experimental results showed that the

performance of our proposed method was improved in both benign and adversarial environments.

2. Importance of smart meters data processing – case of Saudi Arabia, Authors: T. Alquthami, A. M. Alsubaie, and M. Anwer

Description: This paper presents a thorough analysis of 30-minute data sets of KSA residential digital meters to identify all possible discrepancies in the data sets and devise statistical techniques best suited to remove these discrepancies as per the nature of each discrepancy. The analysis is performed through a program that was developed in Python-Pandas. The program parses through three month's meter measurements of 3,283 consumers throughout KSA and detects data inconsistencies, duplicates, missing and outlier values and other issues in the data sets. Statistical techniques that are part of the program are then implemented to correct for these issues. A validation process was developed and included in the program to ensure the adjustment process produces the best reliable outcomes. Analysis indicates that smart meters data have issues that need preprocessing to be used for other applications. The outcome of the program developed shows that smart meters measurement outcome data set could be considered as a valid and trusted.

III. SYSTEM ANALYSIS

3.1 EXISTING SYSTEM:

- Abdallah et al. introduced a review to investigate different approaches for uncovering fraudulent activities in the health care domain based on statistical approaches.
- Popatand Chaudhary presented an extensive review work on credit card fraud detection .The authors provide a detailed analysis of various ML classification methods with their methodology and challenges.
- Ryman-Tubb et al. reviewed several state-of-the-art methods for detecting payment card fraudulent activities using transactional volumes. The study showed that only eight approaches have a practical implication to be used in the industry.
- A study by Albashrawi and Lowell analyzed several studies for one decade covering fraud detection in financial sectors using data mining techniques. However, this was not exhaustive and comprehensive enough as they ignored the method of evaluations and the pros and cons of data mining techniques, among others.

DISADVANTAGES OF EXISTING SYSTEM:

- The existing system model with Logistic regression fails to predict a continuous outcome.
- The existing system model with Logistic regression may not be accurate if the sample size is too small.
- The existing system may lead to overfitting problem.
- The existing system accuracy depends on the quality of the data.
- With large data, the prediction stage might be slow.
- Sensitive to the scale of the data and irrelevant features.
- Require high memory – need to store all of the training data.

3.2.1.ADVANTAGES:

➢ It will give better accuracy

➢ Better accuracy

➢ Better prediction

3.3.SYSTEM REQUIREMENTS

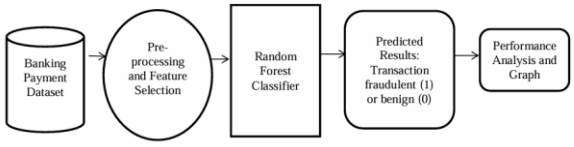
3.3.1.HARDWARE REQUIREMENTS(minimum):

- System : Pentium IV 2.4 GHz
- Hard Disk : 40 GB
- Ram : 512 Mb. 3.3.2.

SOFTWARE REQUIREMENTS:

- Operating System: Windows
- Coding Language: Python 3.7

IV. SYSTEM ARCHITECTURE



V. SYSTEM DESIGN

DATA FLOW DIAGRAM:

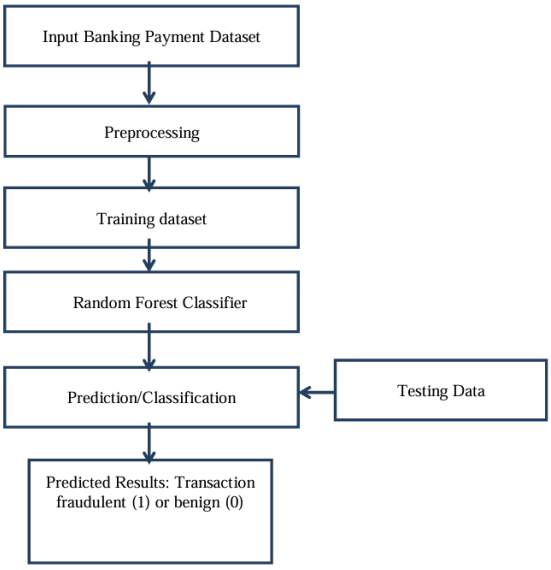
1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.

2. 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.

3. 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.

4. 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.



VI. SOFTWARE ENVIRONMENT

What is Machine Learning : Before we take a look at the details of various machine learning methods, let's start by looking at what machine learning is, and what it isn't. Machine learning is often categorized as a subfield of artificial intelligence, but I find that categorization can often be misleading at first brush. The study of machine learning certainly arose from research in this context, but in the data science application of machine learning methods, it's more helpful to think of machine learning as a means of building models of data. Fundamentally, machine learning involves building mathematical models to help understand data. "Learning" enters the fray when we give these models tunable parameters that can be adapted to observed data; in this way the program can be considered to be "learning" from the data. Once these models have been fit to previously seen data, they can be used to predict and understand aspects of newly observed data. I'll leave to the reader the more philosophical digression regarding the extent to which this type of mathematical, model-based "learning" is similar to the "learning" exhibited by the human brain. Understanding the problem setting in machine learning is essential to using these tools effectively, and so we will start with some broad categorizations of the types of approaches we'll discuss here. Machine learning is used in many different applications, from image and speech recognition to natural language processing, recommendation systems, fraud detection, portfolio optimization, automated task, and so on. Machine learning models are also used to power autonomous vehicles, drones, and robots, making them more intelligent and adaptable to changing environments.

VII. SYSTEM IMPLEMENTATION

Sample code:

```
import numpy as np

import pandas as pd

from flask import Flask, request, jsonify, render_template, redirect, flash, send_file

from sklearn.preprocessing import MinMaxScaler

from werkzeug.utils import secure_filename

import pickle

import numpy as np

import pandas as pd

from flask import Flask, request, jsonify, render_template, redirect, flash, send_file

from sklearn.preprocessing import MinMaxScaler

from werkzeug.utils import secure_filename

import pickle

import numpy as np

import pandas as pd

from sklearn.tree import DecisionTreeClassifier

from sklearn.svm import SVC

app = Flask(__name__) #Initialize the flask App

fraud = pickle.load(open('fraud.pkl','rb'))

@app.route('/')
```

```
@app.route('/first')

def first():

return render_template('first.html')
```

VIII SYSTEM TESTING

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 tests. 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.

IX SCREENSHOTS

INDEX PAGE:

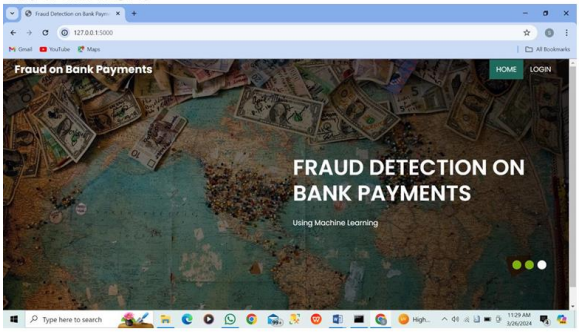
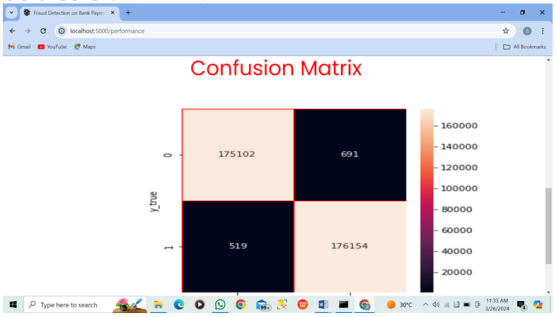


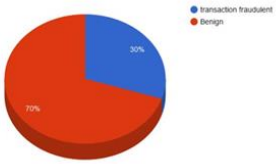
Fig 9.1: Index Page

LOGIN PAGE:

Fig 9.2: Login Page

CONFUSION MATRIX:





CONCLUSION

Financial fraud can occur in a variety of financial contexts, including the corporate, banking, insurance, and taxes sectors. Financial fraud has recently raised concerns among businesses and industries. Financial fraud continues to exist despite several attempts to eradicate it, which has a negative impact on society

and the economy because daily losses from fraud amount to very huge sums of money. Machine-learning-based technologies can now be used intelligently to identify fraudulent transactions by examining a significant amount of financial data, thanks to the development of artificial intelligence. In this article, we published a study that thoroughly analyzed and summarized the body of knowledge on ML-based fraud detection. This study uses the Random Forest Classifier methodology, which extracts, synthesis, and reports results using well-defined methods.

REFERENCES

[1] S. Delecourt and L. Guo, "Building a robust mobile payment fraud detection system with adversarial examples," in 2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), pp. 103–106, IEEE, 2019.

[2] T. Alquthami, A. M. Alsubaie, and M. Anwer, "Importance of smart meters data processing – case of Saudi Arabia," in 2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA), pp. 1–5, IEEE, 2019.

[3] O. Adepoju, J. Wosowei, S. Lawte, and H. Jaiman, "Comparative evaluation of credit card fraud detection using machine learning techniques," in 2019 Global Conference for Advancement in Technology (GCAT), pp. 1– 6, IEEE, 2019.

[4] S. Khatri, A. Arora, and A. P. Agrawal, "Supervised machine learning algorithms for credit card fraud detection: A comparison," in 2020 10th International Conference on Cloud Computing, Data Science Engineering (Confluence), pp. 680–683, IEEE, 2020.

[5] V. Jain, M. Agrawal, and A. Kumar, "Performance analysis of machine learning algorithms in credit cards fraud detection," in 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), pp. 86 88, IEEE, 2020.

[6] Thennakoon, C. Bhagyani, S. Premadasa, S. Mihiranga, and N. Ku- ruwitaarachchi, "Real-time credit card fraud detection using machine learning," in 2019 9th International Conference on Cloud Computing, Data Science Engineering (Confluence), pp. 488–493, IEEE, 2019.