Credit Card Fraud Detection Website Using Machine Learning

A Project Work - I Report

Submitted in partial fulfillment of requirement of the

Degree of

BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE & ENGINEERING

BY

Aaditya Anand – EN19CS301004
Abhay Singh Chauhan – EN19CS301011
Advait Patidar – EN19CS301027

Under the Guidance of
Dr. Harsh Pratap Singh

Department of Computer Science & Engineering
Faculty of Engineering
MEDI-CAPS UNIVERSITY, INDORE- 453331

<u>Report Approval</u>

The project work **"Credit Card Fraud Detection Website Using Machine Learning"** is hereby approved as a creditable study of an engineering/computer application subject carried out and presented in a manner satisfactory to warrant its acceptance as prerequisite for the Degree for which it has been submitted.

It is to be understood that by this approval the undersigned do not endorse or approved any statement made, opinion expressed, or conclusion drawn there in; but approve the "Project Report" only for the purpose for which it has been submitted.

Internal Examiner

Name:

Designation

Affiliation

External Examiner

Name:

Designation

Affiliation

Declaration

We hereby declare that the project entitled **"Credit Card Fraud Detection Website using Machine Learning"** submitted in partial fulfillment for the award of the degree of Bachelor of Technology 'Computer Science & Engineering' completed under the supervision of **Dr. Harsh Pratap Singh,** Faculty of Engineering, Medi-Caps University Indore is an authentic work.

Further, we declare that the content of this Project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for the award of any degree or diploma.

Aaditya Anand

Abhay Singh Chauhan

Advait Patidar

<u>Certificate</u>

I, **Dr. Harsh Pratap Singh** certify that the project entitled **"Credit Fraud Detection Website Using Machine Learning"** submitted in partial fulfillment for the award of the degree of Bachelor of Technology by **Aaditya Anand, Abhay Singh Chauhan** and **Advait Patidar** is the record carried out by them under my guidance and that the work has not formed the basis of award of any other degree elsewhere.

_____   _____

Dr. Harsh Pratap Singh   Dr. Pramod S. Nair

Computer Science & Engineering   Head of the Department

Medi-Caps University, Indore   Computer Science & Engineering

                                          Medi-Caps University, Indore

Acknowledgements

Name: Aaditya Anand, Abhay Singh Chauhan, Advait Patidar
B.Tech. IV Year
Department of Computer Science & Engineering
Faculty of Engineering
Medi-Caps University, Indore

Executive Summary

Nowadays the usage of credit cards has dramatically increased. As credit card becomes the most popular mode of payment for both online as well as regular purchase, cases of fraud associated with it are also rising. Here we model the sequence of operations in credit card transactions processing using a Logistic Regression Model and show it can be used for the detection of frauds. A Logistic Regression Model is initially trained with the normal behavior of a cardholder. If an incoming credit card transaction is not accepted by the trained Logistic Regression model with sufficiently high probability, it is fraudulent. At the same time, we try to ensure that genuine transactions are not rejected. We present experimental results to show the effectiveness of our approach to and compare it with other techniques available in the literature.

Table of Contents

List of Figures

List of Tables

Chapter-1

Introduction

1.1 Introduction

As the payment method is simplified by the combination of the financial industry and IT technology, the payment method of the consumers is changing from cash payment to electronic payment using a credit card, mobile micropayment, and app card. As a result, the number of cases in which anomalous transactions are attempted by abusing e-banking has increased and financial companies started establishing a Fraud Detection System (FDS) to protect consumers from abnormal transactions. The abnormal transaction detection system aims to identify abnormal transactions with high accuracy by analyzing user information and payment information in real-time. Although FDS has shown good results in reducing fraud, most cases being flagged by this system are False Positives that result in substantial investigation costs and cardholder inconvenience. The possibilities of enhancing the current operation constitute the objective of this research. Based on variations and combinations of testing and training class distributions, experiments were performed to explore the influence of these parameters. In this project, we will investigate the trend of abnormal transaction detection using payment log analysis and data mining and summarize the data mining algorithm used for abnormal credit card transaction detection. We will use python programming with Apache spark for advanced processing of data and high accuracy.

1.2 Literature Review

Millions and billions of people use credit cards for payment in both online and offline transactions, due to the existence of a widespread point of sale (POS). countless transactions occurred per minute everywhere on the planet. The reason behind fraud is the negligence of the user. When a third person steals the most important information about credit cards and user details easily, fraud can be achieved. To detect what type of fraud occurs during a transaction, we need to face several challenges. Fetching that among all the transactions that occurred and which one is real could be a task.

Amongst the standard and very common ways of making payment globally and especially in North America, because of the presence of a far-reaching point of sale. A huge number of individuals around the globe use charge cards to buy products and services by getting credit for a time of half a month. Any helpful framework could be mishandled, and a charge card is no exemption from this. Alongside the ascent of charge card use, extortion is on the ascent.

Monetary Institutions (FIs) endure refined fake exercises and bear many dollar misfortunes every year. Considering statistics [2] frauds account for more than $1 billion every year for Visa and MasterCard around the world.

Credit card companies and their banks attempt to discover better approaches to forestall scams. A portion of the precautionary measures on the cards is magnetic stripes, 3D monograms, and CVC. Credit card companies are likewise taking steps to have an alternative for credit cards such as Smart Cards, be that as it may, given assessments this substitution will be over the top expensive because of the broad POS network in the USA and the gigantic no. of cards available for use in those places. FIS additionally utilizes an assortment of computing mechanisms, such as Neural Networks (NNs), to follow and distinguish dubious exchanges and ban them for additional examination.

Nagi et al. presented intrusion detection frauds using data mining techniques for financial fraud detection. The papers from 49 journals were published in 2018. This paper allows the analysis and is classified into four fraud categories and six data mining techniques [5].

Sanjeev et al. Is a paper that analyzes and classifies fraud type classification and fraudulent transaction frequency and the amount by country through actual credit card fraud transaction data and visually expresses the distribution using a box plot [6].

Michael et al. presented signature-based research on fraud detection and a fraud detection model [7] to provide a comprehensive survey of existing research related to fraud detection and to conduct fraudulent transactions in real-time [7]. In the case of real-time detection, it is necessary to make accurate judgments in an instant and consider the characteristics of the data mining algorithm.

TS Quah et al. Implemented real-time detection by separating detection mechanisms into the initial authentication layer, inspection layer, core layer for risk score evaluation and behavior analysis, and additional review layer using the SOM algorithm [8][9][10].

1.3 Objectives

There is a substantial rise in some technologies like "machine learning, artificial intelligence, deep learning" and other relevant fields of information technology. These technologies help us automate the credit card fraud classification process. Automation saves a huge amount of time and work in detecting the fraudulent transactions present in the dataset.

- The key objective of the Credit card Fraud Detection Website is to improvise the procedure of personal follow-up on many suspicious transactions.

- To discover a path to preprocess the flagged records to recognize the probable genuine entries from the list of genuine/falsified entries.

Here, the volume of needless analysis is decreased leading to significant savings for the financial institutions. Moreover, the current FDS threshold can also be lowered and several fraudulent cases, being missed under this level, can be detected.

As a result, the fraud is discovered earlier, and the overall losses may be reduced. For addressing these challenges, outlier detection, and GBT Classifier is used, i.e., among the very commonly used applications of Machine Learning for addressing pattern recognition and classification problems.

The system will overcome the low accuracy forecast problems, utilize the latest AI methods, reduce false alerts, recognize fraudulent transactions, and attain fast and reliable solutions.

1.4 Methodology

The performance of fraud detection in credit card transactions is greatly affected by the sampling approach on the dataset, selection of variables, and detection technique(s) used. The Credit Card Fraud Detection Problem includes modeling past credit card transactions with the knowledge of the ones that turned out to be a fraud. Hence, we are using the technique of machine learning for fraud detection. In this, we take the real bank dataset and split the dataset into a training set and testing set, and then apply the Logistic Regression method.

Reading the dataset from the file "creditcard.csv" which has 2,84,807 transactions. To get all variables in an equivalent range, we subtract the mean and divide it by the standard deviation such that the distribution of the values is normalized.

Then we perform normalization. To get all variables in an equivalent range, we will subtract the mean and divide it by the standard deviation such that the distribution of the values is normalized.

The plotting of the dataset is done. We will first define some models like the "Logistic Regression, Gaussian Naïve-Bayes, and Decision Tree Classifier", and then loop through a training and testing set. First, we will train the model with the training set and then validate the results with the testing set.

1.5 Problem Statement

In the past recent years, credit card breaches have been trending alarmingly. Therefore, it is necessary to develop credit card fraud detection techniques as the counter measure to combat

illegal activities.

There are lots of issues that make this procedure tough to implement and one of the biggest problems associated with fraud detection is the lack of both the literature providing experimental results and of real-world data for academic researchers to perform experiments on. The reason behind this is the sensitive financial data associated with the fraud that has to be kept confidential for the purpose of customer's privacy. Now, here we enumerate different properties a fraud detection system should have to generate proper results:

- The system should be able to handle skewed distributions, since only a very small percentage of all creditcard transactions is fraudulent

- There should be a proper means to handle the noise. Noise is the errors that is present in the data, for example, incorrect dates. This noise in actual data limits the accuracy of generalization that can be achieved, irrespective of how extensive the training set is.

- Another problem related to this field is overlapping data. Many transactions may resemble fraudulent transactions when they are genuine transactions. The opposite also happens when a fraudulent transaction appears to be genuine.

- The systems should be able to adapt themselves to new kinds of fraud. Since after a while, successful fraud techniques decrease in efficiency since they become well known because an efficient fraudster always find a new and inventive ways of performing his job.

These points direct us to the most important necessity of the fraud detection system, which is, a decision layer. The decision layer decides what action to take when fraudulent behavior is observed considering factors like, the frequency and amount of the transaction.

1.6 Chapter Schema

Chapter 2: System Requirement Specification

This chapter includes the information about system requirements like functional and non – functional requirement and specification of the software like software and hardware specification.

Chapter 3: System Analysis & Design

This chapter includes all the diagrams related to the system for better understanding of the system.

Chapter 4: Implementation

This chapter includes the implementation phase of the system and the proposed methods for the system and screenshots of interfaces.

Chapter 5: Testing

This chapter includes the testing phase of the system. Several test cases are included in this chapter.

Chapter 6: Result and Discussion

This chapter includes results drawn by the system.

Chapter 7: Summary & Conclusion

This chapter includes summary about the project.

Chapter 8: Future Scope

This chapter include the future scope possible in the system.

Chapter – 2
System Requirement Specification

2.1 Proposed System

In proposed System, we are applying Random Forest algorithm, Bernoulli Naive Bayes Model and Logistic Regression algorithm for classification of the credit card dataset. Random Forest is an algorithm for classification and regression. Summarily, it is a collection of decision tree classifiers. Random forest has advantage over decision tree as it corrects the habit of over fitting to their training set. A subset of the training set is sampled randomly so that to train each individual tree and then a decision tree is built, each node then splits on a feature selected from a random subset of the full feature set. Even for large data sets with many features and data instances training is extremely fast in random forest and because each tree is trained independently of the others. The Random Forest algorithm has been found to provide a good estimate of the generalization error and to be resistant to over fitting. We propose a Machine learning model to detect fraudulent credit card activities in online financial transactions. Analyzing fake transactions manually is impracticable due to vast amounts of data and its complexity. However, adequately given informative features, could make it is possible using Machine Learning. To classify fraudulent and legitimate credit card transaction by supervised learning Algorithm such as Random Forest. To help us to get awareness about the fraudulent and without loss of any financially.

2.2 Problem with the initial system

In existing System, research about a case study involving credit card fraud detection, where data normalization is applied before Cluster Analysis and with results obtained from the use of Cluster Analysis and Artificial Neural Networks on fraud detection has shown that by clustering attributes neuronal inputs can be minimized. And promising results can be obtained by using normalized data and data should be MLP trained. This research was based on unsupervised learning. Significance of this paper was to find new methods for fraud detection and to increase the accuracy of results. The data set for this paper is based on real life transactional data by a large European company and personal details in data is kept confidential. Accuracy of an algorithm is around 50%. Significance of this paper was to find an algorithm and to reduce the cost measure. The result obtained was by 23% and the algorithm they find was Bayes minimum risk.

2.3 Advantages of the proposed system

1. Random Forest ranks the importance of variables in a regression or classification problem in a natural way can be done by Random Forest.

2. The 'amount' feature is the transaction amount. Feature 'class' is the target class for the binary classification, and it takes value 1 for positive case (fraud) and 0 for negative case (not fraud).

2.4 System Feasibility

Feasibility study includes consideration of all the possible ways to provide a solution to the given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily done based on the future upcoming requirements.

1. **Economic Feasibility**

- The technology used in the project is decided based on the minimum possible cost factor.

- All hardware and software cost must be borne by the organization.

- Overall, we have estimated that the benefits the organization is going to receive from the proposed system will surely overcome the initial costs and the running cost for system.

2. **Technical Feasibility**
   This includes the study of function, performance and constraints that may affect the ability to achieve an acceptable system. For this feasibility study, we study complete functionality to be provided in the system, as described in the SRS, and checked if everything was possible using different types of frontend and backend platforms.

3. **Operational Feasibility**
   The proposed system is fully GUI based and user friendly. A proper training has been conducted to let the employees know of the essence of the system so that they feel comfortable with the system.

2.5 Hardware Specifications

System Processor: Core i3 / i5 / i7

Hard Disk: 500 GB.

Ram: 4 GB.

2.6 Software Specifications

Operating System: Windows 10 and above

Programming language: Python

Framework: Anaconda

IDE: Jupyter Notebook

DL Libraries: Numpy, Pandas

2.7 Functional Requirements

A Functional Requirement (FR) is a statement that describes the service that the programmer must provide. It refers to a software system or a component of one. A function is nothing more than the inputs, behaviors, and outputs of a software system. A computation, data manipulation, business procedure, user interaction, or any other unique feature that determines what function a system is likely to execute can all be considered. Functional Requirements are also known as Functional Specification in Software Engineering.

2.7.1 Preprocessed Data

Data Preprocessing is a technique that is used to convert the raw data into a clean data set. In other words, whenever the data is gathered from different sources it is collected in raw format which is not feasible for the analysis. Therefore, certain steps are executed to convert the data into a small clean data set. This technique is performed before the execution of Iterative Analysis. The set of steps is known as Data Preprocessing.

Data preprocessing is necessary because of the presence of unformatted real-world data. Mostly real – world data is composed of –

* **Inaccurate data (missing data)** - There are many reasons for missing data such as data is not continuouslycollected, a mistake in data entry, technical problems with biometrics and much more.
* **Inconsistent data -** The presence of inconsistencies is due to the reasons such that existence of duplication within data, human data entry, containing mistakes in codes or names, i.e., violation of data constraints and much more.

2.7.2 Training and Testing of Data

The data we use is usually split into training data and test data. The training set contains a known output, and the model learns on this data to be generalized to other data later on. We have the test dataset (or subset) to test our model's prediction on this subset.

2.7.3 Model Creation

The process of training an ML model involves providing an ML algorithm (that is, the learning algorithm) with training data to learn from. The term ML model refers to the model artifact that is created by the training process. The training data must contain the correct answer, which is known as a target or target attribute. The learning algorithm finds patterns in the training data that map the input data attributes to the target (the answer that you want to predict), and it outputs an ML model that captures these patterns. You can use the ML model to get predictions on new data for which you do not know the target. For example, let's say that you want to train an ML model to predict if an email is spam or not spam. You would provide training data that contains emails for which you know the target (that is, a label that tells whether an email is spam or not spam). Machine would train an ML model by using this data, resulting in a model that attempts to predict whether new email will be spam or not spam.

In our project we are using Logistic Regression Algorithm to build our Model on Credit Card Fraud Dataset.

2.8 Non – Functional Requirements

These are basically the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements.

- **Efficiency in Computation:** While the model is executing, the software shall utilize less system's CPU resource and very little of system memory.
- **Extensibility:** The software shall be extensible to support future developments and add-ons.
- **Portability:** The Credit Card Fraud Detection website should be portable to all operating platforms.
- **Performance:** The software shall minimize the number of calculations needed to perform image processing and hand gesture detection.
- **Reliability:** The software shall be operable in all conditions.

Chapter – 3
System Analysis & Design

3.1 System Architecture

A system architecture is the conceptual model that defines the structure, behavior and more vies of a system. An architecture description and representation of a system organized in a way that supports reasoning about the structures and behaviors of the system.
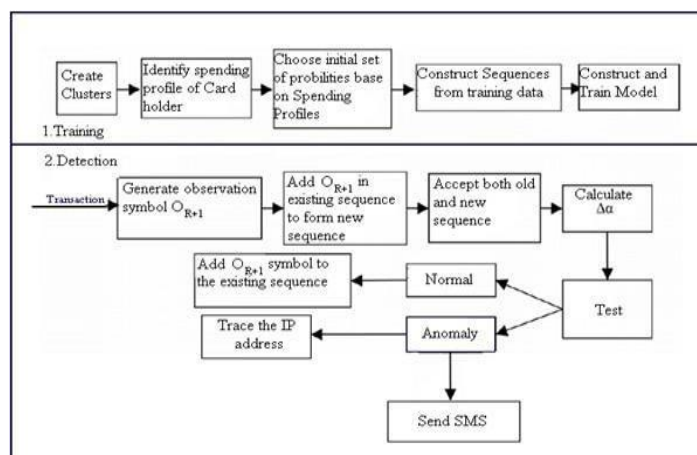

Figure 1: System Architecture of Credit Card Fraud Detection Website

3.2 Data Flow Diagram

A data flow diagram is a way of representing a flow of data through a process or system. The DFD also provides information about the outputs and inputs of each entity and process itself.
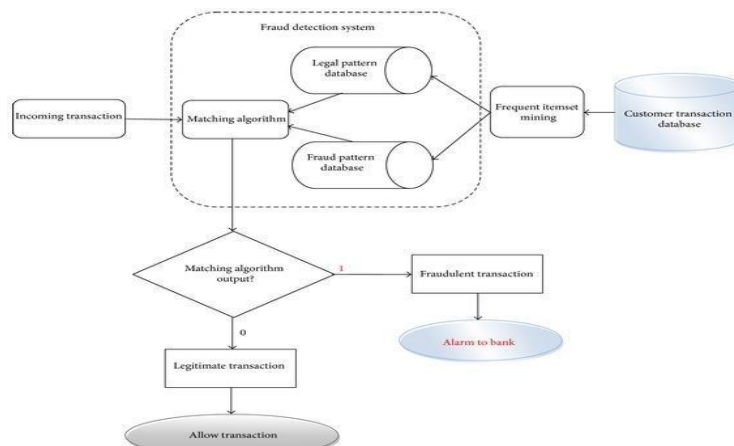

Figure 2: Data Flow Diagram of Credit Card Fraud Detection Website

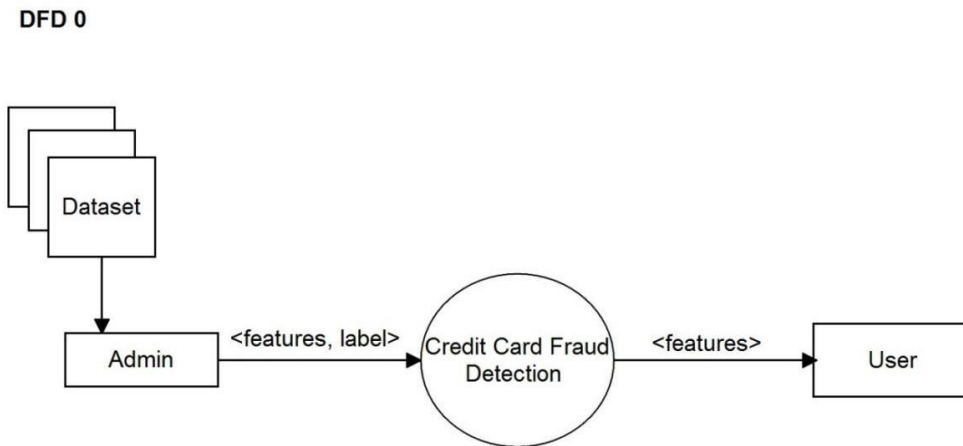### 3.2.1 Data Flow Diagram for Extraction

Figure 3: Data Flow Diagram for extraction of Credit Card Fraud Detection Website

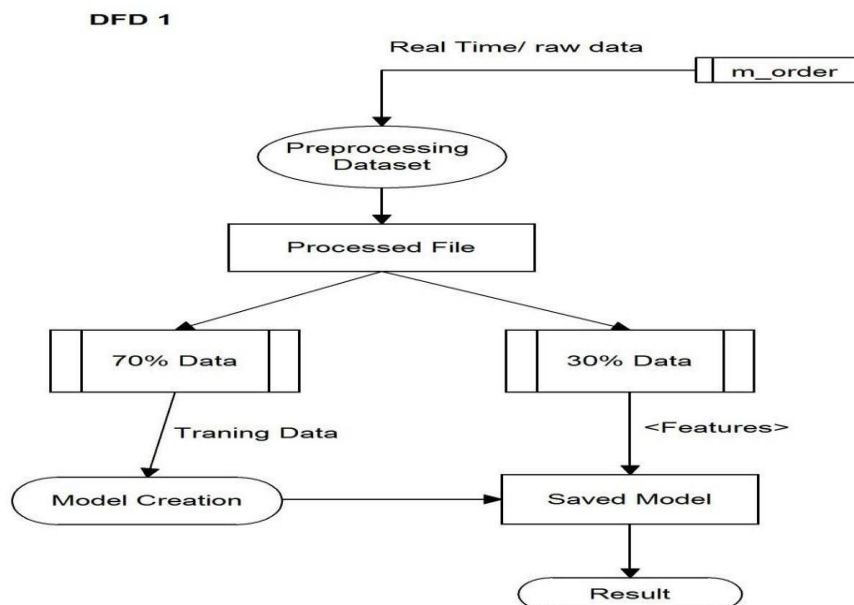### 3.2.2 Data Flow Diagram for Classification of Data

Figure 4: Data Flow Diagram for classification of Credit Card Fraud Detection Website

3.3 Use Case Diagram

A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and tells how the user handles a system.
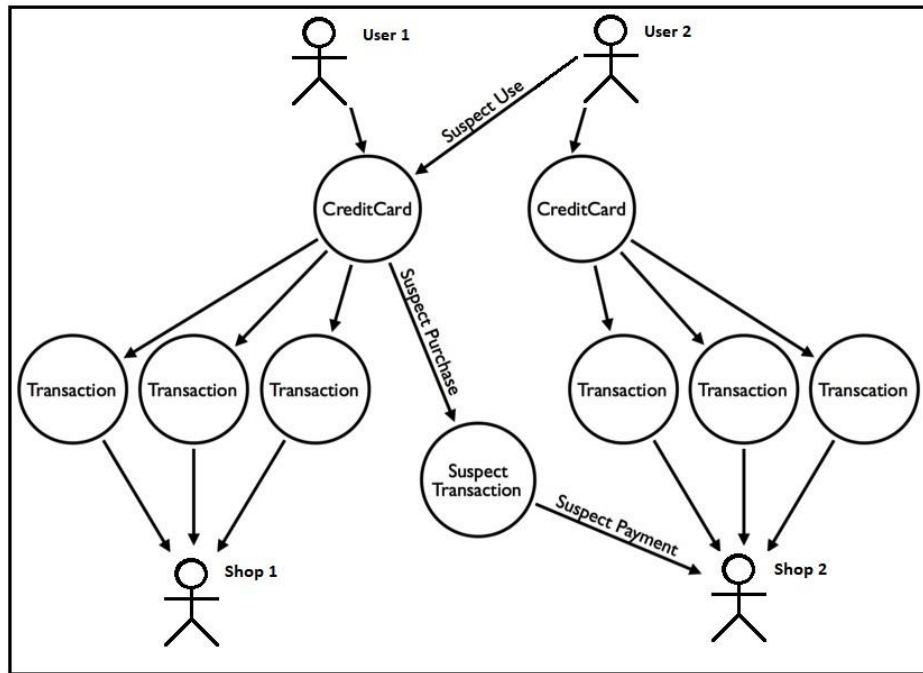
14

Figure 5: Use Case Diagram of Credit Card Fraud Detection Website

Chapter – 4

Implementation

4.1 Procedural Description

The procedure that we propose uses the latest machine learning algorithms to detect anomalous activities, called outliers. The basic rough architecture diagram can be represented with the following figure:
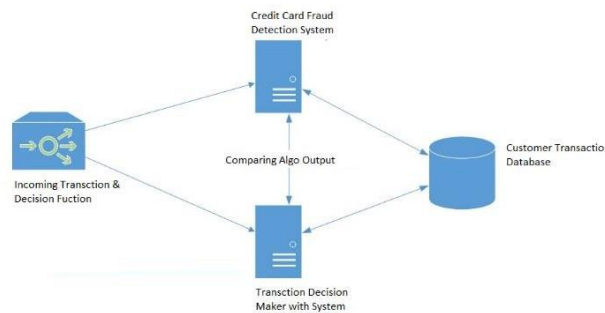

Figure 6: System Architecture with outliers

When looked at in detail on a larger scale along with real life elements, the full architecture diagram can be represented as follows:
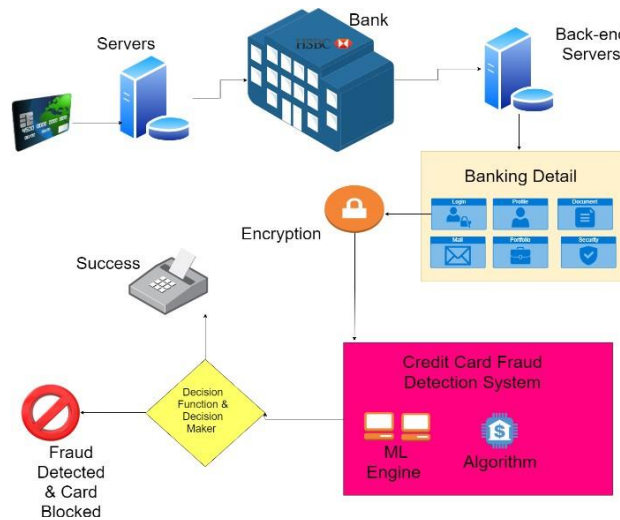
Figure 7: Complete System Diagram

First, we obtained our dataset from Kaggle, a data analysis website which provides datasets. Inside this dataset, there are 31 columns out of which 28 are named as v1-v28 to protect sensitive data. The other columns represent Time, Amount and Class. Time shows the time gap between the first transaction and the following one. Amount is the amount of money transacted. Class 0 represents a valid transaction and 1 represents a fraudulent one. We plot different graphs to check for inconsistencies in the dataset and to visually comprehend it:

Figure 8: Count of Fraudulent vs Non-Fraudulent Transactions

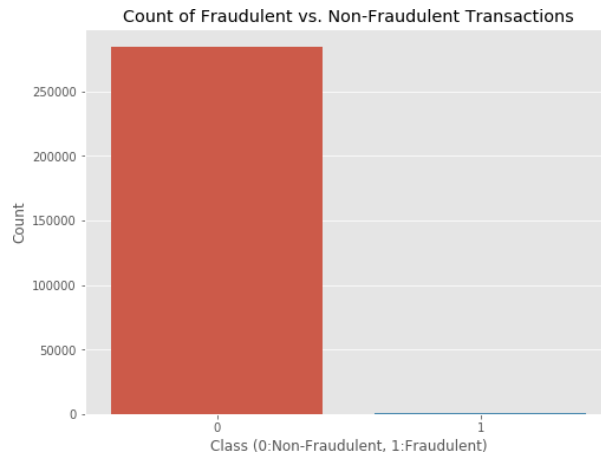Balancing data Since the data is very imbalanced, we will be using some Under sampling and Oversampling techniques. As the name suggests, under sampling is use to reduce the samples from majority class and Oversampling is used to increase the samples from minority class. This way, we can achieve some balancing of the data. Scaling features Now, even though almost all the features are dimensionally reduced using some dimensionality reduction technique, two of the features are in their original form. Time and Amount are the two features which we will be scaling to make our model learn features correctly. Splitting the data now, since we needed to save some entries from the data for our testing purpose, we will now be splitting the data into two parts, namely, train and test. Miscellaneous Now, other than the above three techniques, we did some Exploratory Data Analysis [EDA] on the data, we get the idea of outliers in the data, feature importance etc. by doing this amazing part. One can find the EDA in the notebook itself.

This graph shows the times at which transactions were done within two days. The least number of transactions were made during nighttime and highest during the days.
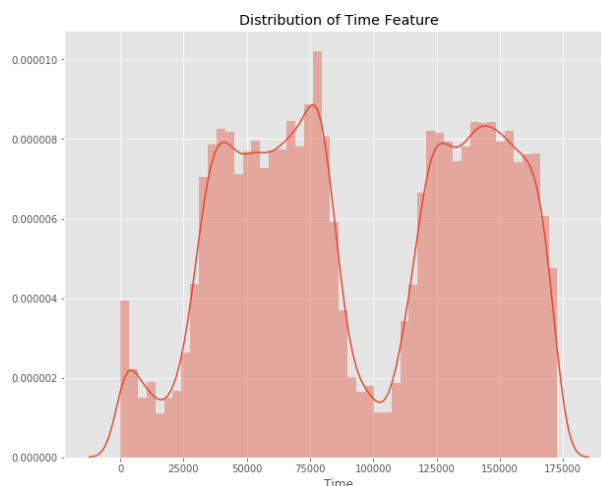

Figure 9: Distribution of Time Features

17

This graph represents the amount that was transacted. Most transactions are relatively small and only a handfulof them come close to the maximum transacted amount. After checking this dataset, we plot a histogram for every column. This is done to get a graphical representation of the dataset which can be used to verify that there are no missing any values in the dataset. This is done to ensure that we don't require any missing value imputation and the machine learning algorithms can process the dataset smoothly.

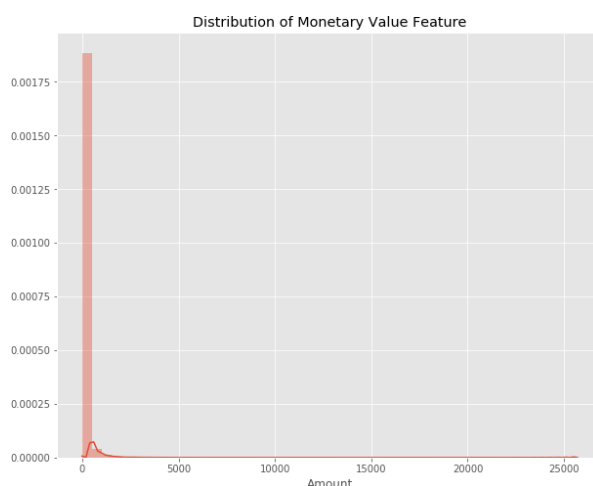*Figure 10: Distribution of Monetary Value Features*

After this analysis, we plot a heatmap to get a colored representation of the data and to study the correlation betweenout predicting variables and the class variable. This heatmap is shown below:
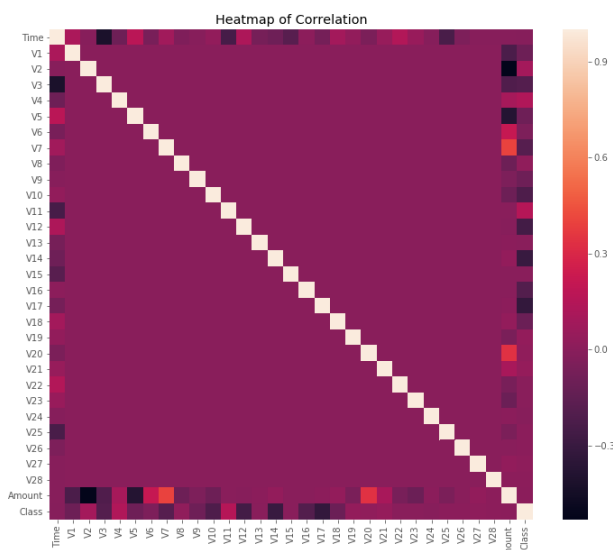

Figure 11: Heatmap of Correlation

4.2 Module Description

The dataset is now formatted and processed. The time and amount column are standardized,

and the Class column is removed to ensure fairness of evaluation. The data is processed by a set of algorithms from modules. This data is fit into a model and the following outlier detection modules are applied on it is Logistic Regression.

These algorithms are a part of sklearn. The ensemble module in the sklearn package includes ensemble-based methods and functions for the classification, regression, and outlier detection.

This free and open-source Python library is built using NumPy, SciPy and matplotlib modules which provides a lot of simple and efficient tools which can be used for data analysis and machine learning. It features various classification, clustering and regression algorithms and is designed to interoperate with the numerical and scientific libraries. We've used Jupyter Notebook platform to make a program in Python to demonstrate the approach that this paper suggests. This program can also be executed on the cloud using Google Collab platform which supports all python notebook files.

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
rf	Random Forest Classifier	0.9996	0.9475	0.7959	0.9547	0.8661	0.8659	0.8705	99.6620
et	Extra Trees Classifier	0.9996	0.9487	0.7783	0.9708	0.8615	0.8613	0.8678	15.9230
catboost	CatBoost Classifier	0.9996	0.9748	0.8018	0.9690	0.8761	0.8759	0.8806	71.3040
knn	K Neighbors Classifier	0.9995	0.9182	0.7702	0.9224	0.8370	0.8367	0.8414	150.9350
xgboost	Extreme Gradient Boosting	0.9995	0.9824	0.7844	0.9444	0.8549	0.8547	0.8594	73.7540
lda	Linear Discriminant Analysis	0.9994	0.9008	0.7667	0.8714	0.8138	0.8135	0.8161	1.0410
lr	Logistic Regression	0.9992	0.9682	0.6187	0.8634	0.7185	0.7181	0.7293	2.3460

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
ada	Ada Boost Classifier	0.9992	0.9663	0.6969	0.8111	0.7481	0.7477	0.7506	30.9640
dt	Decision Tree Classifier	0.9991	0.8731	0.7466	0.7517	0.7462	0.7458	0.7473	10.4550
svm	SVM - Linear Kernel	0.9991	0.0000	0.5545	0.8519	0.6682	0.6678	0.6850	0.4910
gbc	Gradient Boosting Classifier	0.9990	0.6582	0.5182	0.8548	0.6051	0.6047	0.6382	159.5350
ridge	Ridge Classifier	0.9989	0.0000	0.4210	0.8281	0.5554	0.5549	0.5883	0.2740
dummy	Dummy Classifier	0.9983	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2240
lightgbm	Light Gradient Boosting Machine	0.9950	0.6607	0.4545	0.1772	0.2508	0.2489	0.2789	3.8320
nb	Naive Bayes	0.9791	0.9574	0.8223	0.0645	0.1197	0.1168	0.2269	0.3200
qda	Quadratic Discriminant Analysis	0.9765	0.9624	0.8571	0.0600	0.1121	0.1093	0.2232	0.6200

Table 1: Comparison of different models

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
Fold							
0	0.9996	0.9266	0.8000	1.0000	0.8889	0.8887	0.8943
1	0.9998	0.9560	0.8857	1.0000	0.9394	0.9393	0.9410
2	0.9997	1.0000	0.9143	0.9412	0.9275	0.9274	0.9275
3	0.9995	0.9121	0.8000	0.9333	0.8615	0.8613	0.8639
4	0.9995	0.9244	0.7941	0.9310	0.8571	0.8569	0.8596
5	0.9994	0.9241	0.7059	0.9600	0.8136	0.8133	0.8229
6	0.9993	0.9542	0.6765	0.9200	0.7797	0.7793	0.7886
7	0.9997	0.9844	0.8824	0.9375	0.9091	0.9089	0.9094
8	0.9994	0.9537	0.7059	0.9600	0.8136	0.8133	0.8229
9	0.9996	0.9391	0.7941	0.9643	0.8710	0.8708	0.8749
Mean	0.9996	0.9475	0.7959	0.9547	0.8661	0.8659	0.8705

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
Fold							
Std	0.0001	0.0267	0.0775	0.0264	0.0496	0.0497	0.0466

Table 2: Accuracy rating of all the models.

Detailed explanations about the modules with pseudocodes for their algorithms and output graphs are given as follows:

We used Logistic Regression classifier for the project. A logistic regression model is used to predict the probability of a certain class or event existing. We then decide the class from which the entry belongs by using a threshold value. This threshold value is decided by manipulating the Precision-Recall tradeoff.
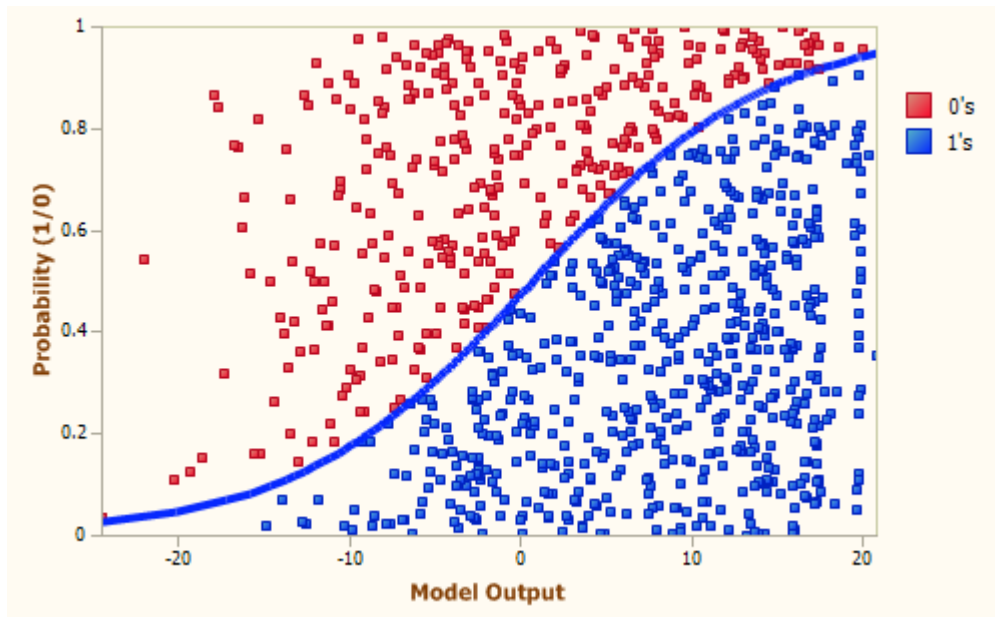

Figure 12: Model Architecture

Also while training, we used GridSearchCV to find the best possible parameters for our model so that it can squeeze the maximum amount of important information out of the data. The pseudo code is as follows:

```python
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LogisticRegression

rng = np.random.RandomState(42)

# Generate train data
X = 0.3 * rng.randn(100, 2)
X_train = np.r_[X + 2, X - 2]
# Generate some regular novel observations
X = 0.3 * rng.randn(20, 2)
X_test = np.r_[X+2, X-2]
# Generate some abnormal novel observations
X_outliers = rng.uniform(low=4, high=4, size=(20, 2))

#fit the model
lr = LogisticRegression(max_iter=1000)
lr.fit(X_train, y_train)

y_pred_train = lr.predict(X_train)
y_pred_test = lr.predict(X_test)
y_pred_outliers = lr.predict(X_outliers)

#plot the line, the samples, and the nearest vectors to the plane
xx, yy = np.meshgrid(np.linspace(-5, 5, 50), np. linspace(-5, 5, 50))
Z = lr.decision_function(np.c_[xx.ravel(), yy.ravel()])
Z = Z.reshape(xx.shape)
```

Figure 13: Pseudo Code

4.3 Creating Web Application

4.3.1 Brief Introduction of a Web Application

A web application is an application software that runs on a web server, unlike computer – based software programs that are stored locally on the Operating System of the device. Web applications are accessed by the user through a web browser with an active internet connection. These applications are programmed using a client-server active internet connection. These applications are programmed using a client-server modeled structure – the user is provided services through an off-site server that is hosted by a third party.

4.3.2 Behind the scenes

We use multiple files to host the web application, the directory structure can be seen as below:
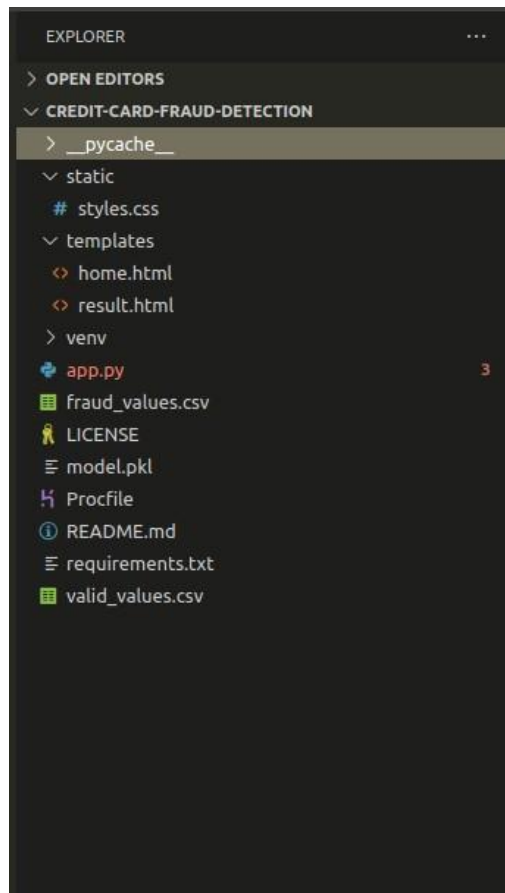
Figure 14: Directory Structure

- App.py is the main application file which will run our app, it will be the first file which will be called and executed, after that, this will render the templates, css styles, and will also take input give by the user.

- Requirements.txt file will contain all the requirements we will be going to use for our work, i.e., mode prediction, reading files, reading models, etc.

- Templates directory contains the html pages we are going to render upon.

- Static contains the css styles which are used for the better formatting of the web app.

- Model.pkl contains the model we got as a result of training which we discussed in the previous section.

- Test data fraud_values.csv and valid_values.csv are the two files which contains the test data regarding the testing of the web app for our users.

- Procfile is the one file which is most important and mandatory if we ant to host our app on Heroku. This file contains the details about the type of app and which file we want to run while starting the application.

4.3.3 How to Predict using the Web Application

We need to copy the data entry which we want to predict into the box. It will take 30 float/integer values as input, with at least one whitespace dividing them.
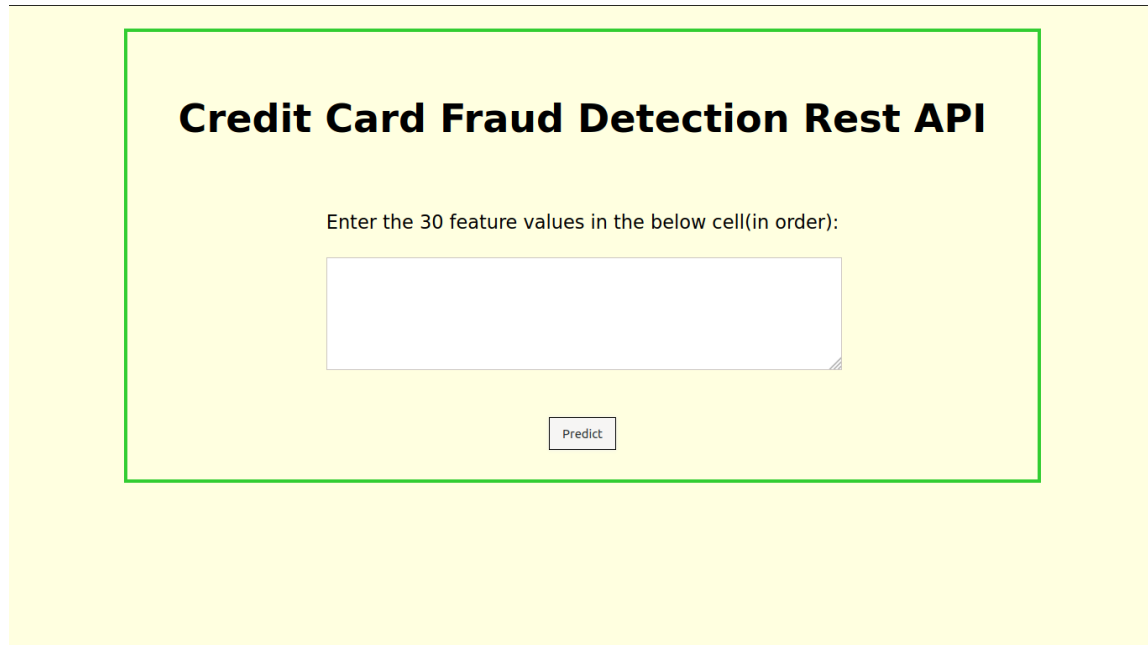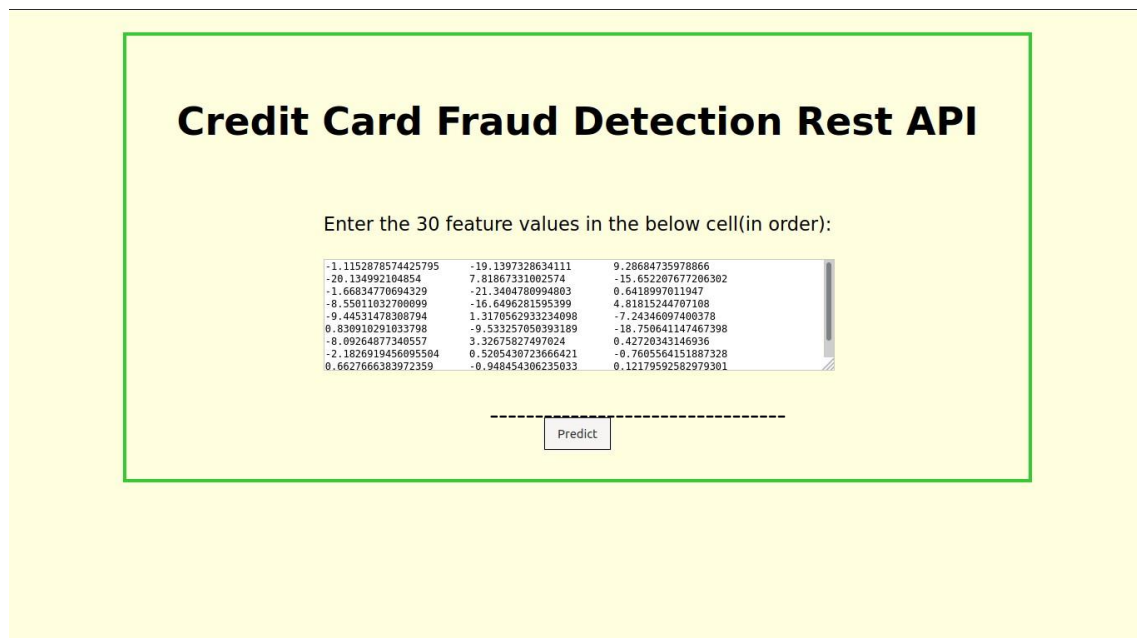

Figure 15: Home Page

Figure 16: Transaction Entry

Credit Card Fraud Detection Results

Validating provided transaction...

According to our model, this transaction is a Fraud transaction.

Figure 17: Fraud Transaction

Credit Card Fraud Detection Results

Validating provided transaction...

According to our model, the provided transaction is not a Fraud transaction.

Figure 18: Valid Transaction

Chapter – 5
Result and Discussion

The code prints out the number of false positives it detected and compares it with the actual values. This is used to calculate the accuracy score and precision of the algorithms. The fraction of data we used for faster testing is 10% of the entire dataset. The complete dataset is also used at the end and both the results are printed.

These results along with the classification report for each algorithm is given in the output as follows, where class 0 means the transaction was determined to be valid and 1 means it was determined as a fraud transaction. This result matched against the class values to check for false positives. Results when 10% of the dataset is used:

Figure 19: Confusion Matrix
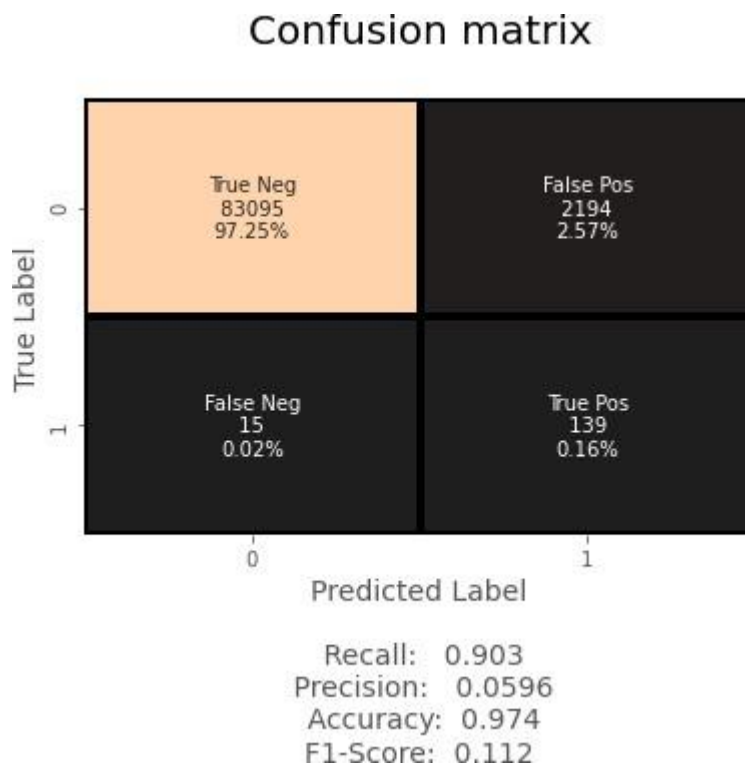
This idea is difficult to implement in real life because it requires the cooperation from banks, which aren't willing to share information due to their market competition, and also due to legal reasons and protection of data of their users.

Therefore, we looked up some reference papers which followed similar approaches and gathered results. As stated in one of these reference papers:

"This technique was applied to a full application data set supplied by a German bank in 2006. For banking confidentiality reasons, only a summary of the results obtained is presented below. After applying this technique, the level 1 list encompasses a few cases but with a high probability of being fraudsters. All individuals mentioned in this list had their cards closed to avoid any risk due to their high-risk profile. The condition is more complex for the other list. The level 2 list is still restricted adequately to be checked on a case-by-case basis.

Credit and collection officers considered that half of the cases in this list could be considered as suspicious fraudulent behavior. For the last list and the largest, the work is equitably heavy. Less than a third of them are suspicious.

To maximize the time efficiency and the overhead charges, a possibility is to include a new element in the query; this element can be the five first digits of the phone numbers, the email address, and the password, for instance, those new queries can be applied to the level 2 list and level 3 list.".

Chapter – 6

Summary and Conclusion

Credit Card is a great tool to pay money easily, but as with all the other monetary payment tools, reliability is a issue here too as it is subjected to breach and other frauds. To encounter this problem, a solution is needed to identify the patterns in the transactions and identify the ones which are fraud, so that finding such transactions beforehand in future will be very easy.

Machine Learning is a great tool to do this work since Machine Learning helps us in finding patterns in the data. Machine Learning can help producing great results if provided enough amount of data. Also, with further advances in the technology, Machine Learning too will advance with time, it will be easy for a person to predict if a transaction is fraud or not much more accurately with the advances.

Chapter – 7

Future Scope

The software has been developed in such a way that it can accept modifications and further changes. The software is very user friendly and future any changes can be done easily. Software restructuring is carried out. Software restructuring modifies source code in an effort to make it amenable to future changes. In general, restructuring does not modify the overall program architecture. It tends to focus on the design details of individual modules and on local data structure defined within modules. Every system should allow scope for further development or enhancement. The system can be adapted for any further development. The system is so flexible to allow any modification need for the further functioning of programs. Since the objectives may not be brought broad in future, the system can be easily modified accordingly, as the system has been modularized. The future expansion can be done in a concise manner in order to improve the efficiently of the system.

While we couldn't reach out goal of 100% accuracy in fraud detection, we did end up creating a system that can, with enough time and data, get very close to that goal. As with any such project, there is some room for improvement here. The very nature of this project allows for multiple algorithms to be integrated together as modules and their results can be combined to increase the accuracy of the final result. This model can further be improved with the addition of more algorithms into it. However, the output of these algorithms needs to be in the same format as the others. Once that condition is satisfied, the modules are easy to add as done in the code. This provides a great degree of modularity and versatility to the project. More room for improvement can be found in the dataset. As demonstrated before, the precision of the algorithms increases when the size of dataset is increased. Hence, more data will surely make the model more accurate in detecting frauds and reduce the number of false positives. However, this requires official support from the banks themselves.

References

[1] *Donald V. Macdougall, Richard G. Mosley, Garioch J. l. Saunders; Credit card crime in Canada: Investigation - Prosecution; The Canadian Association of Crown Counsel; page 1-56; January 1985.*

[2] *Isabelle Sender; Detecting and combating fraud; Chain Store Age; New York; Vol. 74; Issue 7; Page 162; July 1998. International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 07 Issue: 08 | Aug 2020 www.irjet.net p-ISSN: 2395-0072 © 2020, IRJET | Impact Factor value: 7.529 | ISO 9001:2008 Certified Journal | Page 1645*

[3] *Elford Dean, Raj Thomas, Lorry; Visa security center; Personal meetings; January 7 and February 11, 1999.*

[4] *Gyusoo Kim and Seulgi Lee, "2014 Payment Research", Bank of Korea, Vol. 2015, No. 1, Jan. 2015.*

[5] *EWT Nagi, Yong Hu, HY Wong, Yijun Chen, Xin Sun, "The Application of Data Mining Techniques in Financial Fraud Detection: A Classification Framework and an Academic Review of Literature," Decision Support Systems, Vol. 50, No. 3, Feb. 2011.*

[6] *Jha, Sanjeev, J. Christopher Westland, "A Descriptive Study of Credit Card Fraud Pattern," Global Business Review, Vol. 14, No. 3, pp. 373-384, 2015.*

[7] *Edge, Michael Edward, Pedro R. Falcone Sampaio, "A Survey of Signature-based Methods for Financial Fraud Detection," Computers & Security, Vol. 28 No. 6, pp. 381- 394. 2009.*

[8] *Aihua Shen, Rencheng Tong, Yaochen Deng, "Application of Classification Models on Credit Card Fraud Detection," Service Systems and Service Management of the 2007 IEEE International Conference, pp. 1-4, Jun.2007.*

[9] *Chengwei Liu, Yixiang Chan, Syed Hasnain Alam Kazmi, Hao Fu, "Financial Fraud Detection Model: Based on Random Forest," International Journal of Economics and Finance, Vol. 7, No. 7, pp. 178-188, 2015.*

[10] *Ganesh Kumar.None and P.Vasanth Sena, "Novel Artificial Neural Networks and Logistic Approach for Detecting Credit Card Deceit," International Journal of Computer Science and Network Security, Vol. 15, No. 9, Sep. 2015.*

****