A

Mini Project

On

CREDIT CARD FRAUD DETECTION USING FUZZY LOGIC AND NEURAL NETWORK

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

N. SRAVAN KUMAR (217R1A05P3)

B. SHIVA KUMAR (217R1A05L4)

V.ABHIRAM REDDY (217R1A05R5)

Under the Guidance of

Dr. J. NARSIMHARAO

(Associate Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi)
Recognized Under Section 2(f) & 12(B) of the UGC Act.1956,
Kandlakoya (V), Medchal Road, Hyderabad-501401.
2021-25

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled "**CREDIT CARD FRAUD DETECTION USING FUZZY LOGIC AND NEURAL NETWORK** being submitted by **N. SRAVAN KUMAR (217R1A05P3),B.SHIVA KUMAR (217R1A05L4) and V.ABHIRAM REDDY (217R1A05R5)** in partial fulfillment of the requirements for the award of the degree of B. Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by them under our guidance and supervision during the year 2024-25.

The results embodied in this project have not been submitted to any other University or Institute for the award of any degree or diploma.

Dr. J. Narasimharao
Associate Professor
INTERNAL GUIDE

Dr. A. Raji Reddy
DIRECTOR

Dr. Nuthanakanti Bhaskar
HOD

EXTERNAL EXAMINER

Submitted for viva voice Examination held on ________________

ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express our profound gratitude and deep regard to our guide **Dr. J. Narasimharao,** Associate Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carryus a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to Project Review Committee(PRC) Coordinators: **Dr. K. Maheshwari, Dr. J. Narasimharao , Mrs. K. Shilpa, Mr. K. RanjithReddy** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. Nuthanakanti Bhaskar,** Head of the Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy,** Director for being cooperative throughout the course of this project. We would like to express our sincere gratitude to Sri. **Ch. Gopal Reddy,** Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for this constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

N. SRAVAN KUMAR (217R1A05P3)

B. SHIVA KUMAR (217R1A05L4)

V. ABHIRAM REDDY (217R1A05R5)

ABSTRACT

The credit card fraud is mostly come in financial services. The credit card fraud is generated huge number of problems in every year. Lack of research on this credit card problem and submits the real-world credit card fraud analyzes, that is issues. In this project we included best data mining algorithm called “machine learning algorithm”, which is utilized to recognize the credit card fraud, so initially use this algorithm and it is one of the standard model. Then, secondly apply the hybrid methods namely, “AdaBoost and majority vote method”. Use this model efficacy, which is evaluated, and then use the credit card data set it is publicly available one. The financial institution included true world data set, so it is taking and analyzed. In this robustness algorithm additionally evaluate the noise added data samples. This concept is used in experiment and then produce the result positively indicate the hybrid method, that is majority voting, it provides good accuracy rates in credit card fraud detection

LIST OF FIGURES

LIST OF SCREENSHOTS

TABLE OF CONTENTS

1. INTRODUCTION

1. INTRODUCTION

1.1 PROJECT SCOPE

In this proposed project we designed a protocol or a model to detect the fraud activity in credit card transactions. This system is capable of providing most of the essential features required to detect fraudulent and legitimate transactions.

1.2 PROJECT PURPOSE

The primary purpose of this project is to develop an automated fraud detection system that reduces financial losses due to fraudulent transactions. Increases security and trust in credit card transactions. Minimizes the need for human intervention by automating the detection process. Enhances the accuracy of fraud detection by reducing false positives (where legitimate transactions are flagged) and false negatives (where fraudulent transactions are not detected). Leverages fuzzy logic to deal with the uncertainty in fraud detection (e.g., transactions that don’t clearly fall into "fraud" or "not fraud") and neural networks for pattern recognition and classification.

1.3 PROJECT FEATURES

The credit card fraud detection system will use a fuzzy logic-based rule engine to handle ambiguous situations, such as slightly unusual spending behavior, by applying soft decision- making processes. This allows the system to make more nuanced judgments rather than relying on rigid if-then rules. By integrating a neural network such as a multi-layer perceptron or a convolutional neural network, the system can recognize complex patterns in transaction data, effectively learning from historical fraud cases to identify suspicious activities.

2. SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

SYSTEM ANALYSIS

The system for face detection and recognition in organic video content involves multiple stages of processing and analysis. Initially, the system must handle video input-from various sources, such as live streams, interviews, and sports events, ensuring the data is pre-processed to manage noise, motion, and lighting variations. The core of the system lies in the detection and recognition algorithms that need to identify and track faces in real-time despite challenges like rapid motion, crowd interference, and varying camera angles. The system also incorporates performance metrics, allowing it to provide accurate feedback on how well the recognition algorithms perform under different conditions, including challenging scenarios unique to organic video settings.

2.1 PROBLEM DEFINITION

Billions of dollars of loss are caused every year by the fraudulent credit card transactions. Fraud is old as humanity itself and can take an unlimited variety of different forms. The pwc global economic crime survey of 2017 suggests that approximately 48%of organizations experienced economic crime. Therefore, there is definitely an urge to solve the problem of credit card fraud detection. Moreover, the development of new technologies provides additional ways in which criminals may commit fraud. The use of credit cards is prevalent in modern day society and credit card fraud has been kept on growing in recent years. Hugh Financial losses has been fraudulent affects not only merchants and banks, but also individual person who are using the credits. Fraud may also affect the reputation and image of a merchant causing non-financial losses that, though difficult to quantify in the short term, may become visible in the long period. For example, if a cardholder is victim of fraud with a certain company, he may no longer trust their business and choose a contender.

2.2 EXISTING 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 project 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 project 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

2.2.1 LIMITATIONS OF EXISTING SYSTEM

- In this paper a new collective comparison measure that reasonably represents the gain and the losses due to fraud detection is proposed.
- A cost sensitive method which is based on Bayes minimum risk is presented using to the proposed cost measure

2.3 PROPOSED SYSTEM

In proposed System, we are applying random forest 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.

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

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

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 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

2.4.1 ECONOMICAL FEASIBILITY

2.4.2 TECHNICAL FEASIBILITY

2.4.3 BEHAVIORAL FEASIBILITY

2.4.1 ECONOMIC 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.

2.4.2 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.

2.4.3 BEHAVIORAL 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.

2.5 HARDWARE & SOFTWARE REQUIREMENTS

2.5.1 HARDWARE REQUIREMENTS:

Hardware interfaces specifies the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements.

• Processor	-	Intel
• RAM	-	4GB
• Hard Disk	-	260 GB
• Keyboard		
• Mouse		

2.5.2 SOFTWARE REQUIREMENTS:

Software Requirements specifies the logical characteristics of each interface and the software components of the system. The following are some software requirements

• Operating system	:	Windows 10 or above.
• Languages	:	Python
• Tool	:	Anaconda
• DataBase	:	MYSQL
• Server	:	Flask

3. ARCHITECTURE

3. ARCHITECTURE

3.1 PROJECT ARCITECTURE

This project architecture shows the procedure followed for face detection and recognition for an organic video

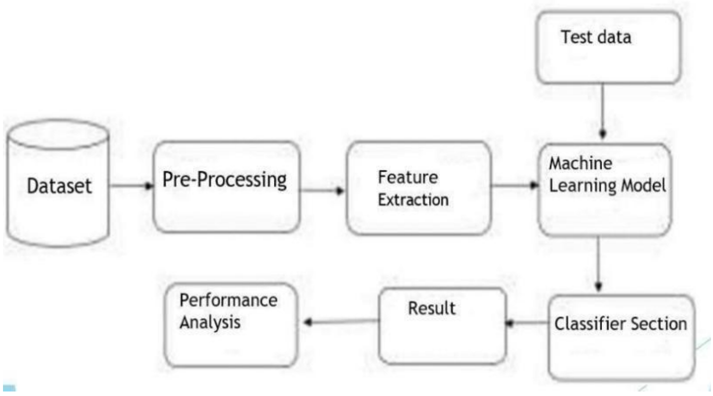


Figure 3.1: Project Architecture of credit card fraud detection using fuzzy logic and neural network

3.2 DESCRIPTION

Dataset: This step consist of a dataset with the full transaction information.
Pre-Processing : Pre-processing refers to the techniques and steps taken to prepare raw data for analysis or modeling
Feature Extraction: This Feature extraction is the process of transforming raw data into a set of usable features that can be effectively used for analysis, modeling, or machine learning tasks. The goal is to reduce the dimensionality of the data while preserving its essential information, making it easier for algorithms to learn patterns.
Test data : in this test data a test data is prepared to classify the transaction is a fraud or not by binary format such as 0's and 1's.
Machine Learning model : In a machine learning model, several key processes and outcomes can occur, depending on the type of model and the data being used such as random forest algo- -rithm . **Classifier Section :** in this the classification is done in a manner that total number of transaction fraud transactions and normal transactions
Result: the results are obtained in a graphical manner.
Performance Analaysis : in this section the performance is analaysed such that we can know the total transactions , normal transactions and fraud transaction.

3.3 USE CASE DIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a visual representation of how a user might interact with a system. It's a type of dynamic diagram that models the behavior of a system and helps capture its requirements

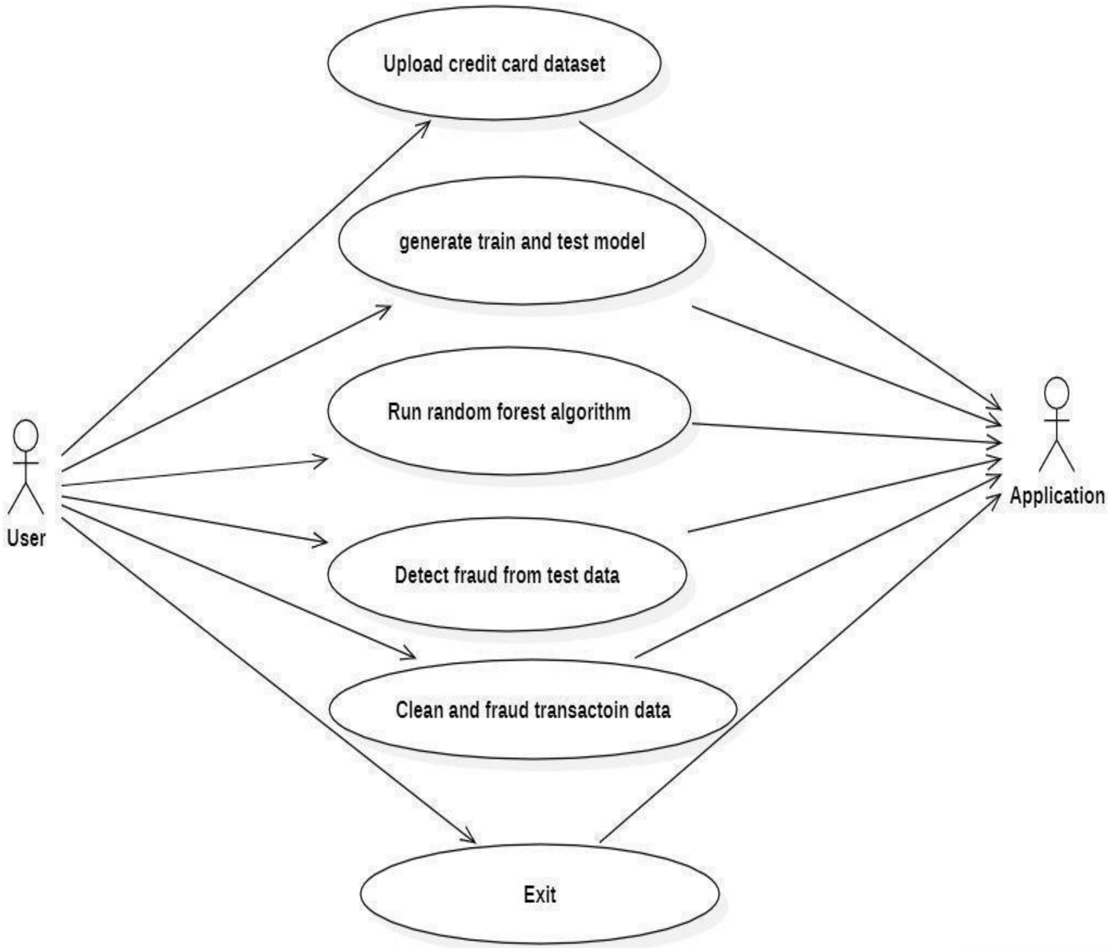

Upload credit card dataset
generate train and test model
Run random forest algorithm
Detect fraud from test data
Clean and fraud transactoin data
Exit
User
Application

Figure 3.2 : Use case diagram for credit card fraud detection using fuzzy logic & neural network

3.4 CLASS DIAGRAM

Class Diagram is a collection of classes and objects.

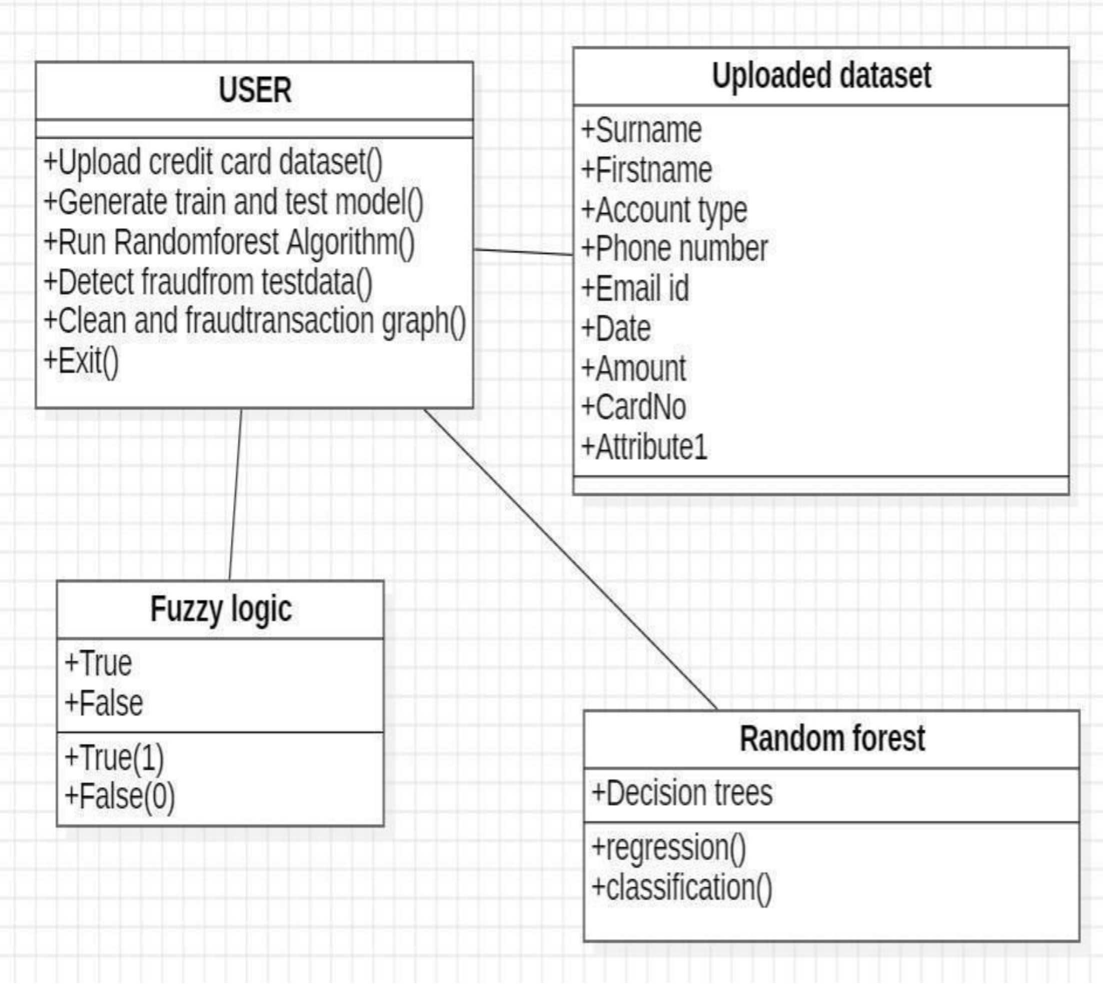


Figure 3.3 : Sequence diagram of Credit Card fraud detection using Fuzzy logic and Neural networks

3.5 SEQUENCE DIAGRAM

It describes about sequence of activity

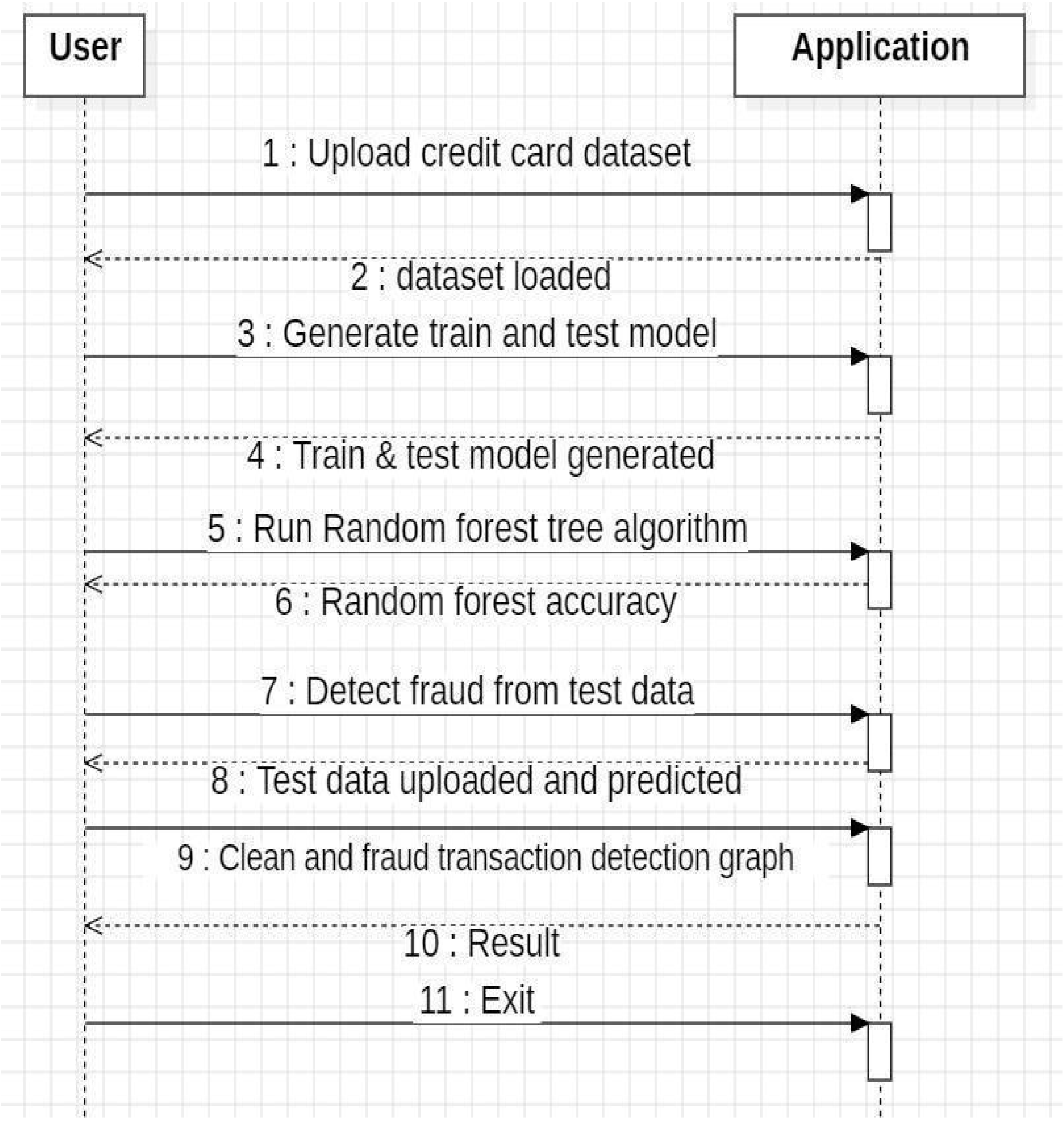

User
Application
1 : Upload credit card dataset
2 : dataset loaded
3 : Generate train and test model
4 : Train & test model generated
5 : Run Random forest tree algorithm
6 : Random forest accuracy
7 : Detect fraud from test data
8 : Test data uploaded and predicted
9 : Clean and fraud transaction detection graph
10 : Result
11 : Exit

Figure 3.4 : Sequence diagram of Credit Card fraud detection using Fuzzy logic and Neural networks

3.6 ACTIVITY DIAGRAM

It describes about flow of activity states.

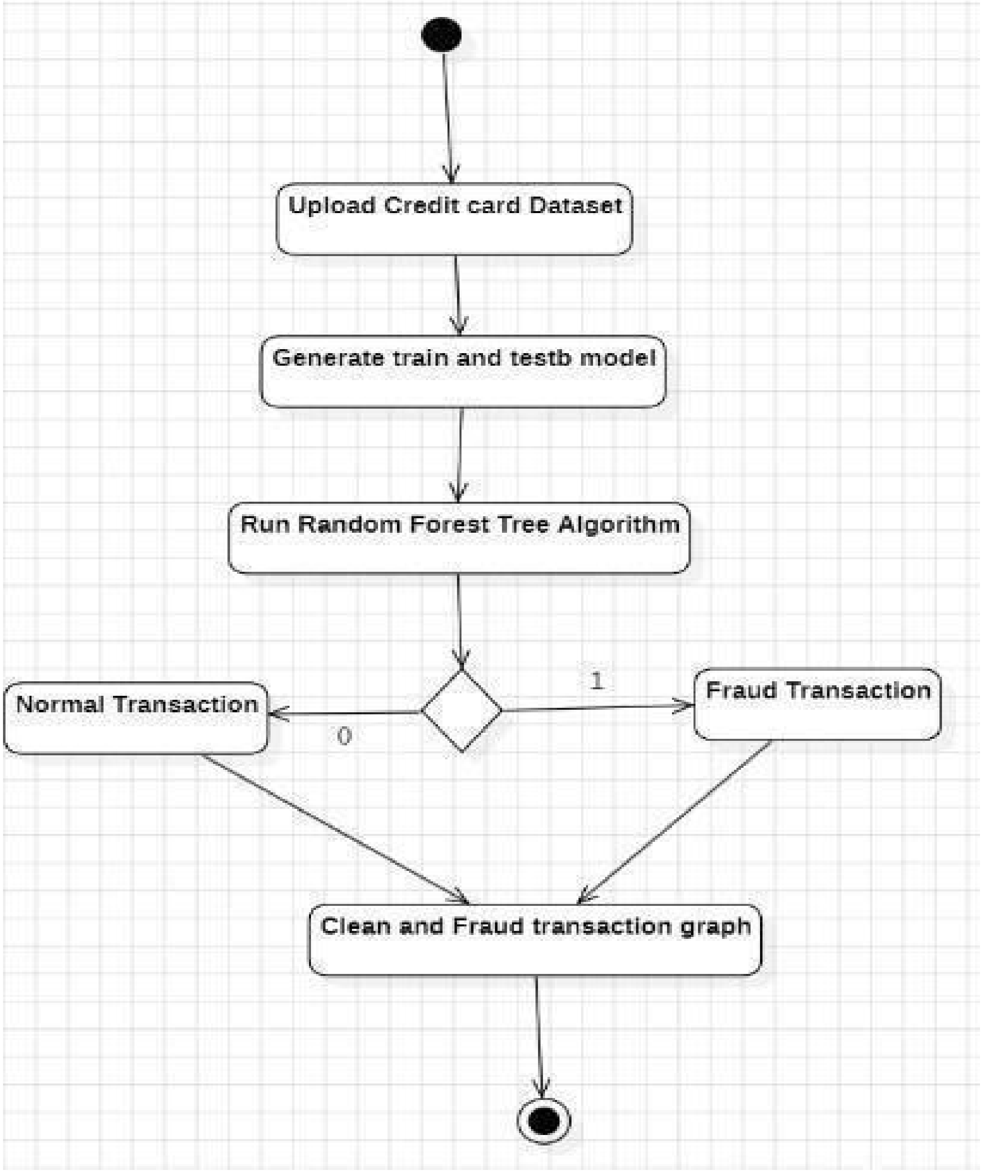

Upload Credit card Dataset
Generate train and testb model
Run Random Forest Tree Algorithm
Normal Transaction
0
1
Fraud Transaction
Clean and Fraud transaction graph

Figure 3.5 : Activity diagram of Credit Card fraud detection using Fuzzy logic and Neural networks

4. IMPLEMENTATION

4. IMPLEMENTATION

4.1 SAMPLE CODE

```
from tkinter import
messagebox from tkinter
import *
from tkinter import
simpledialog import
tkinter
from tkinter import
filedialog import
matplotlib.pyplot as plt
import numpy as np
from tkinter.filedialog import
askopenfilename import numpy as np
import pandas
as pd from
sklearn import *
from sklearn.model_selection import
train_test_split  from  sklearn.metrics
import accuracy_score
from sklearn.metrics import classification_report
from sklearn.ensemble import
RandomForestClassifier #from sklearn.tree
import export_graphviz
#from IPython import display
main = tkinter.Tk()
main.title("Credit Card Fraud Detection") #designing
main screen main.geometry("1300x1200")
global filename
global cls
global X, Y, X_train, X_test, y_train, y_test

global random_acc # all global variables names define in
above lines global clean
```

```
    global attack
    global total

def traintest(train): #method to generate test and train data from
    dataset X = train.values[:, 0:29]
    Y = train.values[:, 30]
    print(X)
    print(Y)

    X_train, X_test, y_train, y_test =
    train_test_split(X, Y, test_size = 0.3,
    random_state = 0)
    return X, Y, X_train, X_test, y_train, y_test

def generateModel(): #method to read dataset values which contains all five
    features data global X, Y, X_train, X_test, y_train, y_test
    train = pd.read_csv(filename)

    X, Y, X_train, X_test, y_train, y_test = traintest(train)
    text.insert(END,"Train & Test Model Generated\n\n")
    text.insert(END,"Total Dataset Size : "+str(len(train))+"\n")
    text.insert(END,"Split Training Size :
    "+str(len(X_train))+"\n") text.insert(END,"Split Test Size :
    "+str(len(X_test))+"\n")

def upload(): #function to upload tweeter
    profile global filename
    filename = filedialog.askopenfilename(initialdir="dataset"

   text.delete('1.0', END)
    text.insert(END,filename+" loaded\n");
```

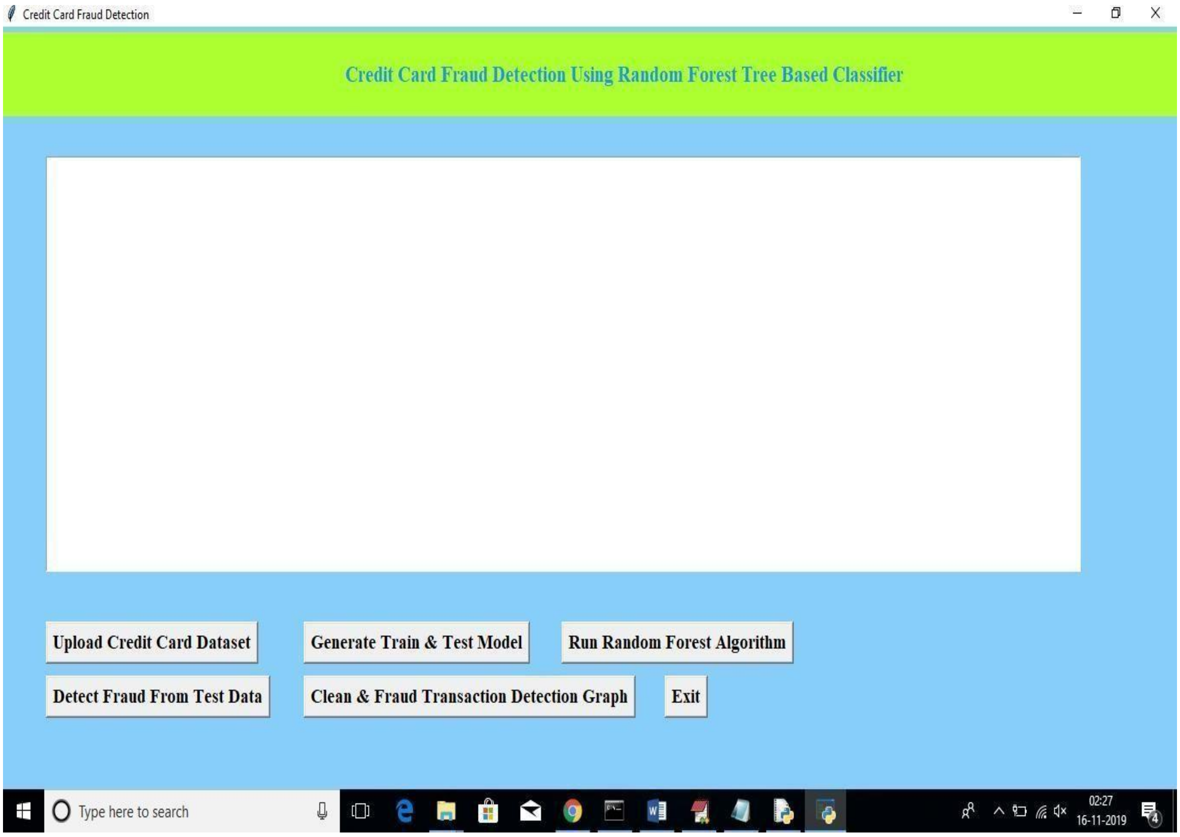
```
  def prediction(X_test, cls): #prediction done here
 y_pred =
  cls.predict(X_test) for i
  in range(50):
   print("X=%s, Predicted=%s" % (X_test[i],
  y_pred[i])) return y_pred

  # Function to calculate accuracy


def cal_accuracy(y_test, y_pred, details):
  accuracy =
  accuracy_score(y_test,y_pred)*100
  text.insert(END,details+"\n\n")
  text.insert(END,"Accuracy :
  "+str(accuracy)+"\n\n") return accuracy


def runRandomForest():
  headers =
  ["Time","V1","V2","V3","V4","V5","V6","V7","V8","V9","V10","V11","V12","V13","V14","
  V15","V16","V1 7",
  "V18","V19","V20","V21","V22","V23","V24","V25","V26","V27","V28","Amount","Class"]
  global random_acc
  global cls
  global X, Y, X_train, X_test, y_train, y_test
  cls =
  RandomForestClassifier(n_estimators=50,max_depth=2,random_state=0,class_weight='balanc
  ed') cls.fit(X_train, y_train)
  text.insert(END,"Prediction Results\n\n")
  prediction_data = prediction(X_test, cls)
  random_acc = cal_accuracy(y_test, prediction_data,'Random Forest Accuracy')


    #str_tree = export_graphviz(cls, out_file=None, feature_names=headers,filled=True,
  special_characters=True, rotate=True, precision=0.6)
  #display.display(str_tree)
```

```
def predicts():
   global clean
   global attack
   global total
   clean = 0;
   attack = 0;
   text.delete('1.0',
   END)
   filename                                        =
   filedialog.askopenfilename(initialdir="dataset") test
   = pd.read_csv(filename)
   test = test.values[:, 0:29]
   total = len(test)
   text.insert(END,filename+" test file
   loaded\n"); y_pred = cls.predict(test)
   for i in range(len(test)):

if str(y_pred[i]) == '1.0':
   attack = attack + 1
   text.insert(END,"X=%s, Predicted = %s" % (test[i], 'Contains Fraud TransactionSignature')+"\n\n")else:
   clean = clean + 1

   text.insert(END,"X=%s, Predicted = %s" % (test[i], 'Transaction Contains Cleaned Signatures')+"\n\n")



   def graph():

   height = [total,clean,attack]

   bars  =  ('Total  Transactions','Normal  Transaction','Fraud
   Transaction') y_pos = np.arange(len(bars))
   plt.bar(y_pos,
   height)
   plt.xticks(y_pos,
   bars) plt.show()
```

```
font = ('times', 16, 'bold')

title = Label(main, text='Credit Card Fraud Detection Using Random Forest Tree Based Classifier')

title.config(bg='greenyellow', fg='dodger blue')

title.config(font=font)

title.config(height=3,

width=120)

title.place(x=0,y=5)


font1 = ('times', 12, 'bold')

text=Text(main,height=20,width=15

0) scroll=Scrollbar(text)

text.configure(yscrollcommand=scroll.

set) text.place(x=50,y=120)

text.config(font=font1)


font1 = ('times', 14, 'bold')

uploadButton = Button(main, text="Upload Credit Card Dataset",

command=upload) uploadButton.place(x=50,y=550)

uploadButton.config(font=font1)


modelButton = Button(main, text="Gene
rate Train & Test Model", command=generateModel)
```

5. SCREENSHOTS

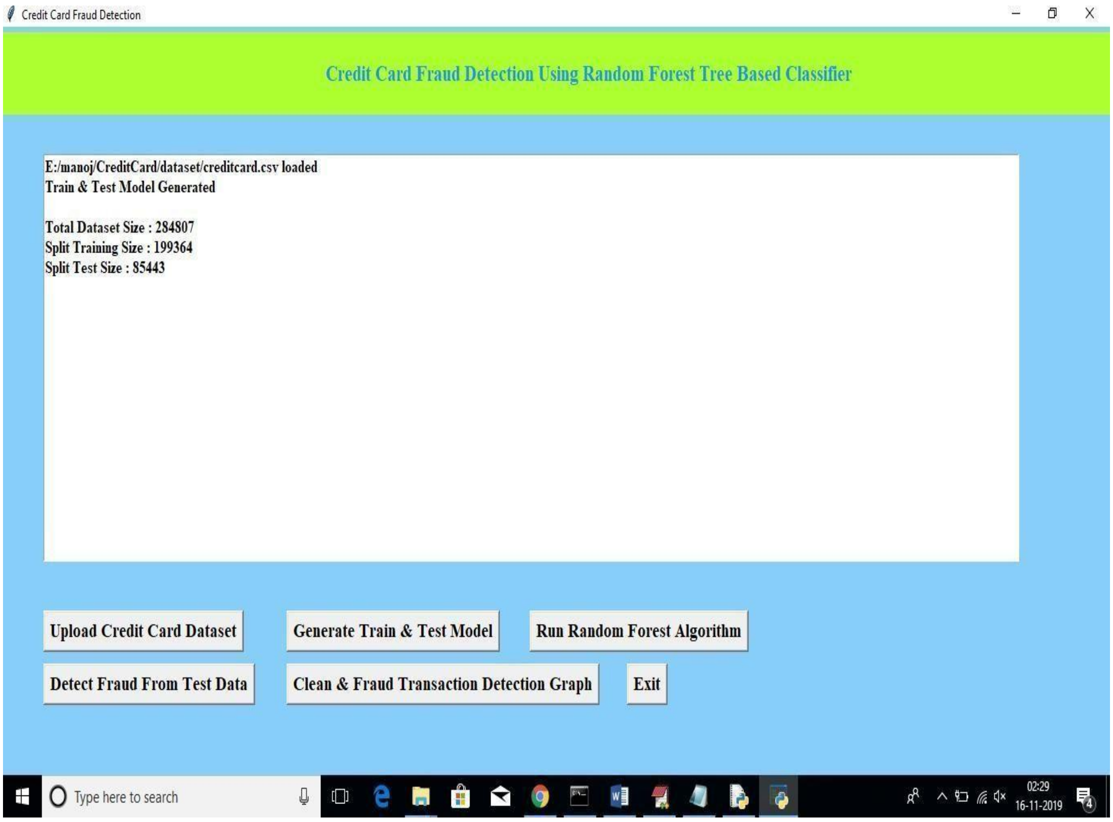
5.1 RESULT OF CREDIT CARD FRAUD DETECTION


Credit Card Fraud Detection
Credit Card Fraud Detection Using Random Forest Tree Based Classifier
Upload Credit Card Dataset
Generate Train & Test Model
Run Random Forest Algorithm
Detect Fraud From Test Data
Clean & Fraud Transaction Detection Graph
Exit
Type here to search
02:27
16-11-2019

Screenshot 5.1 : Result of credit card fraud detection

5.2 Algorithm Result

we can see total records available in dataset and then application using how many records for training and how many for testing



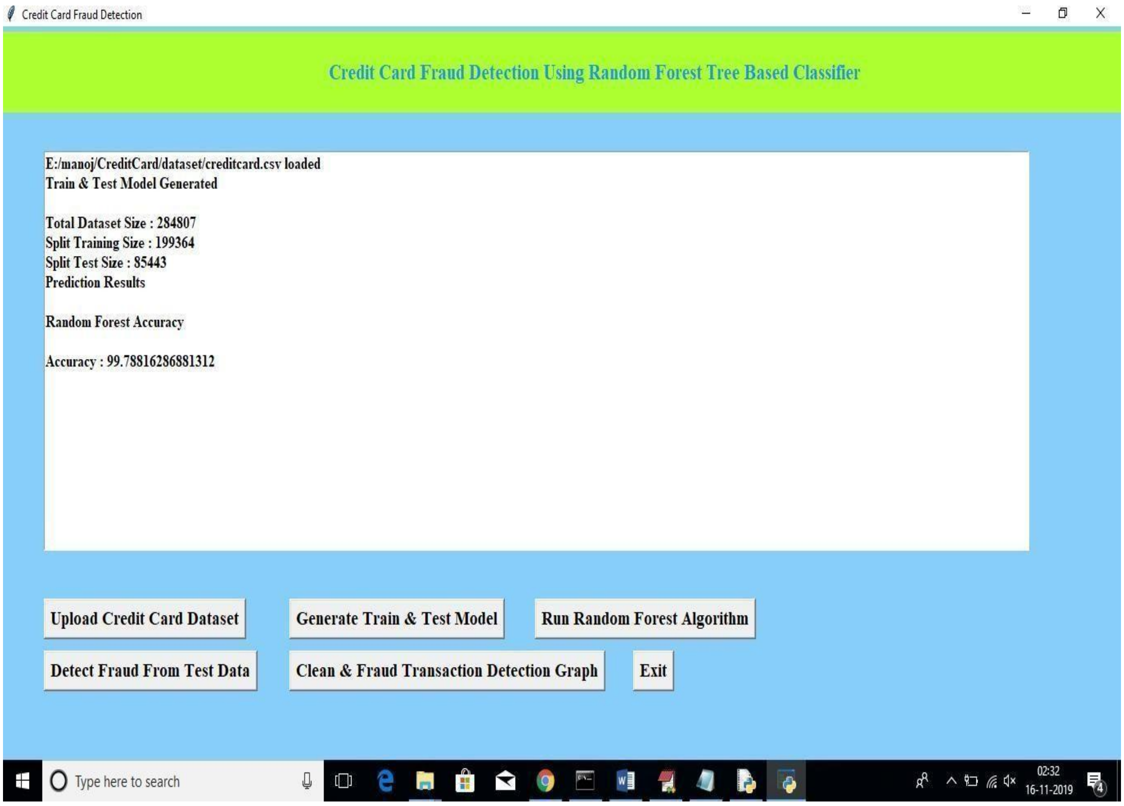
```
E:/manoj/CreditCard/dataset/creditcard.csv loaded
Train & Test Model Generated

Total Dataset Size : 284807
Split Training Size : 199364
Split Test Size : 85443
```

Credit Card Fraud Detection
Credit Card Fraud Detection Using Random Forest Tree Based Classifier
Upload Credit Card Dataset
Generate Train & Test Model
Run Random Forest Algorithm
Detect Fraud From Test Data
Clean & Fraud Transaction Detection Graph
Exit
Type here to search
02:29
16-11-2019

Screenshot 5.2 : Algorithm Result

5.3 : Accuracy Result

Random Forest generate 99.78% percent accuracy while building model ontrain and test data

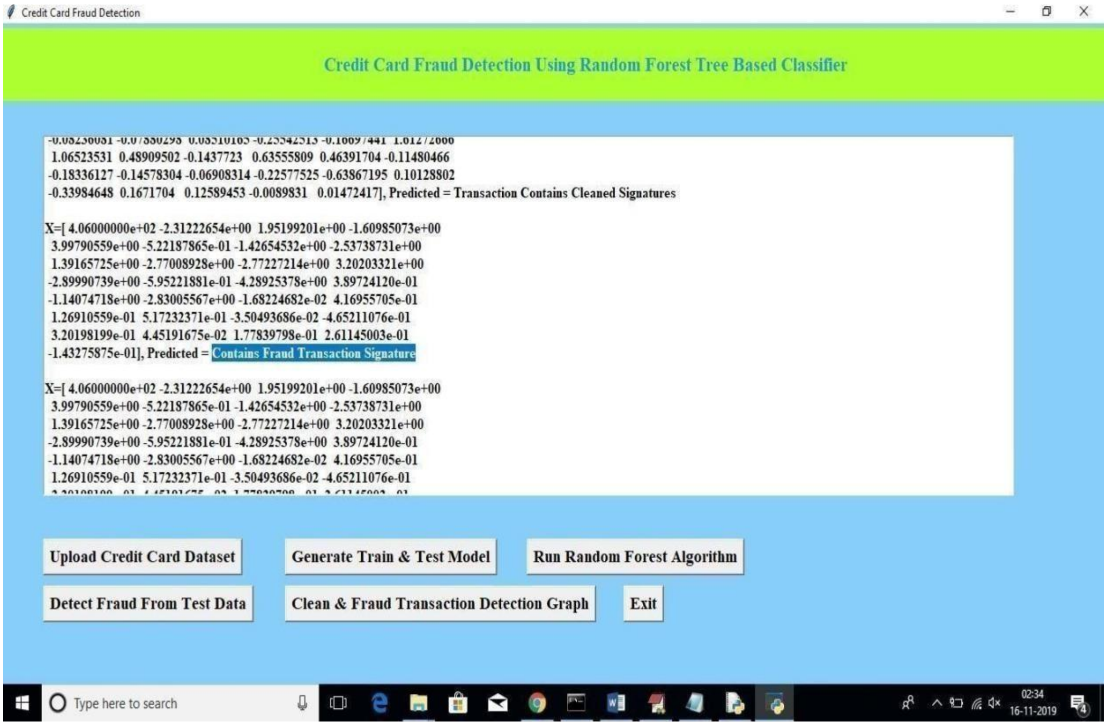

Credit Card Fraud Detection
Credit Card Fraud Detection Using Random Forest Tree Based Classifier
Upload Credit Card Dataset
Generate Train & Test Model
Run Random Forest Algorithm
Detect Fraud From Test Data
Clean & Fraud Transaction Detection Graph
Exit
Type here to search
02:32
16-11-2019

```
E:/manoj/CreditCard/dataset/creditcard.csv loaded
Train & Test Model Generated

Total Dataset Size : 284807
Split Training Size : 199364
Split Test Size : 85443
Prediction Results

Random Forest Accuracy

Accuracy : 99.78816286881312
```

Screenshot 5.3 : Accuracy Result.

5.4 : Final Result

Each test data application will display output as whether transaction contains cleaned or fraud signatu

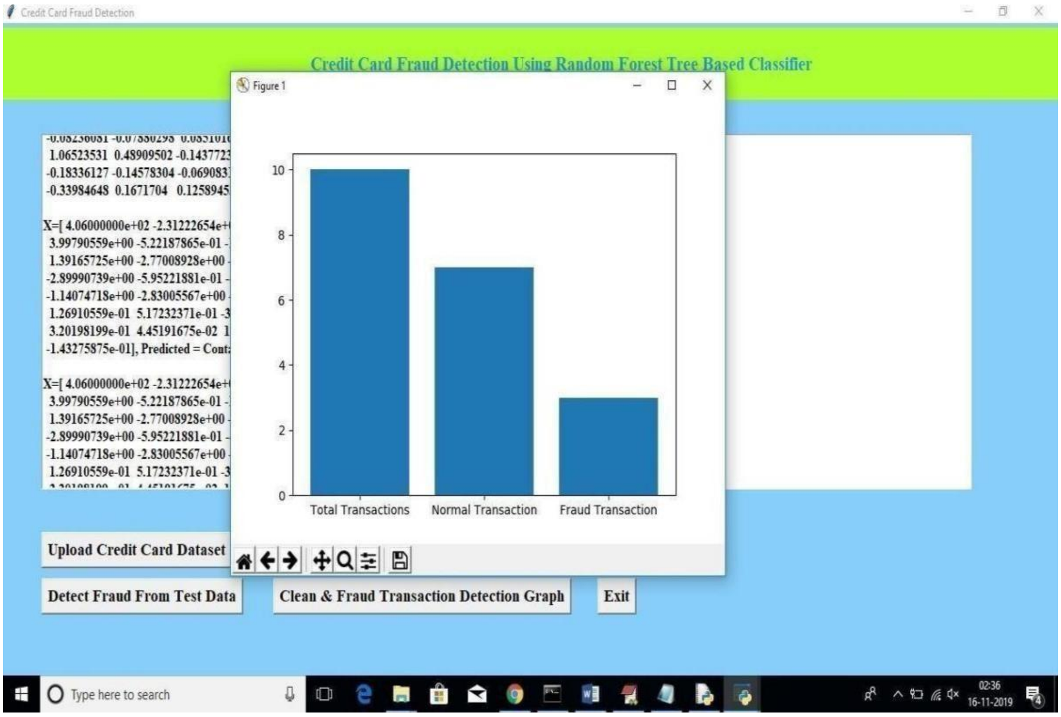

Credit Card Fraud Detection
Credit Card Fraud Detection Using Random Forest Tree Based Classifier
Upload Credit Card Dataset
Generate Train & Test Model
Run Random Forest Algorithm
Detect Fraud From Test Data
Clean & Fraud Transaction Detection Graph
Exit
Type here to search
02:34
16-11-2019

```
-0.08236081 -0.07880298 0.08510165 -0.25542513 -0.16697441 1.61272666
 1.06523531 0.48909502 -0.1437723  0.63555809 0.46391704 -0.11480466
-0.18336127 -0.14578304 -0.06908314 -0.22577525 -0.63867195 0.10128802
-0.33984648 0.1671704  0.12589453 -0.0089831  0.01472417], Predicted = Transaction Contains Cleaned Signatures

X=[ 4.06000000e+02 -2.31222654e+00 1.95199201e+00 -1.60985073e+00
 3.99790559e+00 -5.22187865e-01 -1.42654532e+00 -2.53738731e+00
 1.39165725e+00 -2.77008928e+00 -2.77227214e+00 3.20203321e+00
-2.89990739e+00 -5.95221881e-01 -4.28925378e+00 3.89724120e-01
-1.14074718e+00 -2.83005567e+00 -1.68224682e-02 4.16955705e-01
 1.26910559e-01 5.17232371e-01 -3.50493686e-02 -4.65211076e-01
 3.20198199e-01 4.45191675e-02 1.77839798e-01 2.61145003e-01
-1.43275875e-01], Predicted = Contains Fraud Transaction Signature

X=[ 4.06000000e+02 -2.31222654e+00 1.95199201e+00 -1.60985073e+00
 3.99790559e+00 -5.22187865e-01 -1.42654532e+00 -2.53738731e+00
 1.39165725e+00 -2.77008928e+00 -2.77227214e+00 3.20203321e+00
-2.89990739e+00 -5.95221881e-01 -4.28925378e+00 3.89724120e-01
-1.14074718e+00 -2.83005567e+00 -1.68224682e-02 4.16955705e-01
 1.26910559e-01 5.17232371e-01 -3.50493686e-02 -4.65211076e-01
 3.20198199e-01 4.45191675e-02 1.77839798e-01 2.61145003e-01
```

Screenshot 5.4 : Final Result

5.5 : Final Output

we can see total test data and number of normal and fraud transaction detected**.**



Credit Card Fraud Detection
Credit Card Fraud Detection Using Random Forest Tree Based Classifier
Figure 1
10
8
6
4
2
0
Total Transactions
Normal Transaction
Fraud Transaction
Upload Credit Card Dataset
Detect Fraud From Test Data
Clean & Fraud Transaction Detection Graph
Exit
Type here to search
02:36
16-11-2019

```
-0.08236081 -0.07880298 0.0831010
 1.06523531  0.48909502 -0.1437723
-0.18336127 -0.14578304 -0.0690833
-0.33984648  0.1671704   0.1258945

X=[ 4.06000000e+02 -2.31222654e+
 3.99790559e+00 -5.22187865e-01 -
 1.39165725e+00 -2.77008928e+00 -
-2.89990739e+00 -5.95221881e-01 -
-1.14074718e+00 -2.83005567e+00 -
 1.26910559e-01  5.17232371e-01 -3
 3.20198199e-01  4.45191675e-02  1
-1.43275875e-01], Predicted = Cont

X=[ 4.06000000e+02 -2.31222654e+
 3.99790559e+00 -5.22187865e-01 -
 1.39165725e+00 -2.77008928e+00 -
-2.89990739e+00 -5.95221881e-01 -
-1.14074718e+00 -2.83005567e+00 -
 1.26910559e-01  5.17232371e-01 -3
```

Screenshot 5.5 : Final Output

6. TESTING

6. TESTING

6.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover very 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.

6.2 TYPES OF TESTING

6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic 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.

6.2.2 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.

6.2.3 FUNCTIONAL TESTING

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.
System/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.

6.3 TEST CASES

6.3.1 UPLOADING IMAGES

Test case ID	Test case name	Purpose	Test Case	Output
1	User uploads credit dataset	Use it for identification	The user uploads the credit card details.	Uploaded successfully
2	User click generate train & test model	Use it for identification	The user uploads the dataset	Uploaded successfully

6.3.2 CLASSIFICATION

Test case ID	Test case name	Purpose	Input	Output
1	User perform random forest algorithm	To check if the classifier performs its task	A transaction dataset is given	Data is classified.
2	Detect fraud from test data	To check if the classifier performs its task	Detecting by 0 and 1	It predicted as fraud or not.
3	Clean & fraud transaction graph	To check if the classifier performs its task	Total result data	Bar graph is visualised

7. CONCLUSION

7. CONCLUSION & FUTURE SCOPE

7.1 PROJECT CONCLUSION

The Random forest algorithm will perform better with a larger number of training data, but speed during testing and application will suffer. Application of more pre-processing techniques would also help. The SVM algorithm still suffers from the imbalanced data set problem and requires more pre-processing to give better results at the results shown by svm is great but it could have been better if more pre-processing have been done on the data.

The constraints are met and overcome successfully. The system is designed as like it was decided in the design phase. The project gives good idea on developing a full-fledged application satisfying the user requirements.

The system is very flexible and versatile. Validation checks induced have greatly reduced errors. Provisions have been made to upgrade the software. The application has been tested with live data and has provided a successful result. Hence the software has proved to work efficiently

7.2 FUTURE SCOPE

In future we can use other convolutional neural networks by downloading the modules directly into the project files. The software can be developed further to include lot of modules because the proposed system is developed on the view of future. We can connect to other data bases by including them .

8. BIBLIOGRAPHY

8. BIBLIOGRAPHY

8.1 REFERENCES

1. Sudhamathy G: Credit Risk Analysis and Prediction Modelling of Bank Loans Using R,vol. 8, No-5 , pp. 1954-1966.
2. LI Changjian, HU Peng: Credit Risk Assessment for ural Credit Cooperatives based on Improve Neural Network, International Conference on Smart Grid and Electrical Automation vol. 60, no - 3, pp 227-230, 2017.
3. Wei Sun, Chen-Guang Yang, Jian-Xun Qi: Credit RiskAssessment in Commercial Banks Based On Support Vector Machines, vol.6, pp 2430-2433, 2006.
4. Amlan Kundu, Suvasini Panigrahi, Shamik Sural, Senior Member, IEEE, “BLAST-SSAHAHybridization for Credit Card Fraud Detection”, vol. 6, no. 4 pp. 309-315, 2009.
5. Y. Sahin and E. Duman, “Detecting Credit Card Fraud by Decision Trees and Support VectorMachines, Proceedings of International Multi Conference of Engineers and Computer Scientists, vol. I, 2011.
6. Sitaram patel, Sunita Gond , “Supervised Machine (SVM) Learning for Credit Card Fraud Detection, International of engineering trends and technology, vol. 8, no. -3, pp. 137-140,2014
7. Snehal Patil, Harshada Somavanshi, Jyoti Gaikwad, Amruta Deshmane, Rinku Badgujar,CreditCard Fraud Detection Using Decision Tree Induction Algorithm, International Journal of Computer Science and Mobile Computing, Vol.4 Issue.4, April-2015, pg. 92-95.
8. Dahee Choi and Kyungho Lee, “Machine Learning based Approach to Financial Fraud Detection Process in MobilePayment System", vol. 5, no. - 4, December 2017, pp. 12-24

8.2 GITHUB LINK

[1] https://web.stanford.edu/class/cs231a/prev_projects_2016/output%20(1).pdf