A PROJECT PHASE - I REPORT ON

“Credit Card Fraud Detection using Deep Learning”

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE

FOURTH YEAR ENGINEERING (INFORMATION TECHNOLOGY)

BY

Ms. Pandharpote Anushka Sunil	**B190108562**
Mr. Rokade Dhananjay Dadasaheb	**B190108570**
Ms. Dumbre Rutuja Sanjay	**B190108521**
Ms. Patil Apeksha Sanjaykumar	**B190108564**

UNDER THE GUIDANCE OF

Prof. S. C. Deshmukh

DEPARTMENT OF INFORMATION TECHNOLOGY

AMRUTVAHINI COLLEGE OF ENGINEERING, SANGAMNER

A/P-Ghulewadi-422608, Tal. Sangamner, Dist. Ahmednagar (MS) India

YEAR 2022-23

DEPARTMENT OF INFORMATION TECHNOLOGY

Amrutvahini College of Engineering, Sangamner

A/P-Ghulewadi-422608, Tal. Sangamner, Dist. Ahmednagar (MS) India

Year 2022-23

CERTIFICATE

This is to certify that Project Phase -I entitled

“Credit Card Fraud Detection using Deep Learning”

Is submitted as partial fulfilment of

curriculum of the B.E. of Information Technology

BY

Ms. Pandharpote Anushka Sunil	**B190108562**
Mr. Rokade Dhananjay Dadasaheb	**B190108570**
Ms. Dumbre Rutuja Sanjay	**B190108521**
Ms. Patil Apeksha Sanjaykumar	**B190108564**

(Prof. S. C. Deshmukh)
Project Guide

(Dr. A. V. Markad)
Project Co-Ordinator

Dr. B. L. Gunjal
Head of Dept.

Dr. M. A. Venkatesh
Principal

Savitribai Phule Pune University

CERTIFICATE

This is to Certify that

Ms. Pandharpote Anushka Sunil	**B190108562**
Mr. Rokade Dhananjay Dadasaheb	**B190108570**
Ms. Dumbre Rutuja Sanjay	**B190108521**
Ms. Patil Apeksha Sanjaykumar	**B190108564**

Students of B.E.I.T.

were examined in Project Phase -I report entitled

“Credit Card Fraud Detection using Deep Learning”

on . . ./. . . /2022

At

DEPARTMENT OF INFORMATION TECHNOLOGY,

AMRUTVAHINI COLLEGE OF ENGINEERING,SANGAMNER

YEAR 2022-23

. . . .

Internal Examiner

. . . .

External Examiner

Certificate By Guide

This is to certify that

Ms. Pandharpote Anushka Sunil	**B190108562**
Mr. Rokade Dhananjay Dadasaheb	**B190108570**
Ms. Dumbre Rutuja Sanjay	**B190108521**
Ms. Patil Apeksha Sanjaykumar	**B190108564**

Has completed the project work under my guidance and that, I have verified the work for its originality in documentation, problem statement, literature survey and conclusion presented in work.

Place: Sangamner

Date:

(Prof. S. C. Deshmukh)
AVCOE,Sangamner

Acknowledgement

We express our sincere thanks to all those who have provided us the valuable guidance towards the successful completion of this system as a part of syllabus for the bachelors course.

We hereby take this opportunity to sincerely thank **Prof. S. C. Deshmukh** for his valuable guidance, inspiration, whole hearted involvement during every stage of this project and his experience, perception through professional knowledge which made it possible for us in successfully realizing the concept.

We would also like to thanks our project co-ordinator **Dr. A. V. Markad** for providing all the assistance and facilities which were vital in completing this dissertation.

We are also thankful to **Dr. B. L. Gunjal** - Head of Department – Information Technology for her constant enlightenment, support and motivation which has been highly instrumental in successful completion of our project phase 1.

We are extremely thankful to **Dr. M. A. Venkatesh** Principal - Amrutvahini College Of Engineering, Sangamner for his encouragement and providing us the opportunity and facilities to carry out this work.

Finally, we like to express our deep sense of gratitude towards our parents, friends and well-wishers who were always there for suggestions and help.

Abstract

People can use credit cards for online transactions as it provides an efficient and easy-to-use facility. With the increase in usage of credit cards, the capacity of credit card misuse has also enhanced. Credit card frauds cause significant financial losses for both credit card holders and financial companies. In this study, the main aim is to detect such frauds, including the accessibility of public data, high-class imbalance data, the changes in fraud nature, and high rates of false alarm. The main focus is to apply Deep Learning algorithm of the LSTM and RNN model by adding the additional layers for features extraction and the classification of credit card transactions as fraudulent or not. The main aim is to detect fraudulent transactions using credit cards with the help of deep learning algorithms. Feature selection algorithms are used to rank the top features from the CCF transaction dataset, which help in class label predictions. The deep learning model is proposed by adding a number of additional layers that are then used to extract the features and classification from the credit card farad detection dataset. To analyze the performance both model, apply different architecture of layers. DL methods, are associated with the processing of text and the baseline model. Using these methods for the detection of credit cards yields better performance than traditional algorithm.

INDEX

List of Figures

Chapter 1

Introduction

Nowadays Credit card usage has been drastically increased across the world, now people believe in going cashless and are completely dependent on online transactions. The credit card has made the digital transaction easier and more accessible. A huge number of dollars of loss are caused every year by the criminal credit card transactions. Fraud is as old as mankind itself and can take an unlimited variety of different forms. The PwC global economic crime survey of 2017 suggests that approximately 48organizations experienced economic crime. Therefore, there's positively a necessity to unravel the matter of credit card fraud detection. Moreover, the growth of new technologies provides supplementary ways in which criminals may commit a scam. The use of credit cards is predominant in modern day society and credit card fraud has been kept on increasing in recent years. Huge Financial losses have been fraudulent effects on not only merchants and banks but also the individual person who are using the credits. Fraud may also affect the reputation and image of a merchant causing non-financial losses that. For example, if a cardholder is a victim of fraud with a certain company, he may no longer trust their business and choose a competitor. Fraud Detection is the process of monitoring the transaction behavior of a cardholder to detect whether an incoming transaction is authentic and authorized or not otherwise it will be detected as illicit.

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. As technology changes, it becomes difficult to track the Modeling and pattern of fraudulent transactions. With the rise of machine learning, artificial intelligence and other relevant fields of information technology, it becomes feasible to automate this process and to save some of the intensive amount of labour that is put into detecting credit card fraud.

1.1 Motivation

- The use of credit cards is predominant in modern day society and credit card fraud has been kept on increasing in recent years.
- Huge Financial losses have been fraudulent effects on not only merchants and banks but also the individual person who are using the credits.
- Fraud may also affect the reputation and image of a merchant causing non-financial losses that.

1.2 Aim

- To design and develop a python based system to predict the credit card fraudulent through machine learning and deep learning model.

1.3 Objectives

1. Design of a protocol or a model to detect the fraud activity in credit card transactions.
2. The system is capable of providing most of the essential features required to detect fraudulent and legitimate transactions.
3. As technology changes, it becomes difficult to track the Modeling and pattern of fraudulent transactions

Chapter 2

Literature Survey

1. ”Fraud Detection Techniques for Credit Card Transactions,”

   The problem of detecting a credit score includes modeling of past transactions for credit cards with the facts of those who have been revealed to fraud. The version is then used to delay whether new transactions are fraudulent or are now not new transactions. Our goal here is to detect 100% of fraudulent transactions and minimize fraudulent misclassification. Credit score card fraud detection is a common class model. In this method, we focused on the analysis and preprocessing of several anomaly detection algorithms and record sets, such as ”neighbor outliers” and ”forest zone isolation” algorithms, in PCA-converted credit card transaction statistics[1].

2. ”Fraud Detection in Credit Card Transactions Using SVM and Random Forest Algorithms,”

   It is critical for all banks that issue credit cards to reduce the cost of implementing an effective fraud detection system. One of the most difficult challenges is that neither the card nor the cardholder is needed to complete the transaction during a credit card transaction. Thus, the seller cannot verify whether the customer who is making an acquisition is an authentic cardholder or not. The accuracy of detecting fraud is improvised with this system proposed using random forest algorithm, decision tree, and support vector machine algorithms. A random forest algorithm is a classification process for observing the data set and optimizing the accuracy of the resultant data. The techniques’ performance is judged based on precision, sensitivity, & accuracy. Some of the data provided are processed to identify fraud detection and provide visualization for the graphic model [2].

3. ”Fraud Detection in Credit Card Data using Unsupervised Machine Learning Based Scheme,”

   Fraud transactions in credit card data transaction are increasing each year. In this direction, researchers are also trying the novel techniques to detect and prevent such frauds. However, there is always a need of some techniques that should precisely and efficiently detect these frauds. This paper proposes a scheme for detecting frauds in credit card data which uses a Neural Network (NN) based unsupervised learning technique. Proposed method outperforms the existing approaches of Auto Encoder (AE), Local Outlier Factor (LOF), Isolation Forest (IF) and K-Means clustering. Proposed NN based fraud detection method performs with 99.87% accuracy whereas existing methods AE, IF, LOF and K Means gives 97%, 98%, 98% and 99.75% accuracy respectively [3].

4. ”Performance Analysis of Machine Learning Algorithms in Credit Cards Fraud Detection,”

   Credit cards are very commonly used in making online payments. In recent years’ frauds are reported which are accomplished using credit cards. It is very difficult to detect and prevent the fraud which is accomplished using credit card. Machine Learning(ML) is an Artificial Intelligence (AI) technique which is used to solve many problems in science and engineering. In this paper, machine learning algorithms are applied on a data set of credit cards frauds and the power of three machine learning algorithms is compared to detect the frauds accomplished using credit cards. The accuracy of Random Forest machine learning algorithm is best as compared to Decision Tree and XGBOOST algorithms [4] .

5. ”Credit Card Fraud Detection Using State-of-the-Art Machine Learning and Deep Learning Algorithms,”

   The relevant literature presents many machines learning based approaches for credit card detection, such as Extreme Learning Method, Decision Tree, Random Forest, Support Vector Machine, Logistic Regression and XG Boost. However, due to low accuracy, there is still a need to apply state of the art deep learning algorithms to reduce fraud losses [5].

Chapter 3

Requirements And Analysis

3.1 Problem Definition

- To design and develop a python based system to predict the credit card fraudulent through machine learning and deep learning model.

3.2 Software Requirement Specification (SRS)

3.2.1 Introduction

This document will provide a general description of project, including user requirements, product perspective, and overview of requirements, general constraints. In addition, it will also provide the specific requirements and functionality needed for this project such as interface, functional requirements and performance requirements.

- **Purpose :** The main purpose for preparing this document is to give a general insight into the analysis and requirements of the existing system or situation and for determining the operating characteristics of the system.
- **Scope :**
  1. Avoid Financial Losses
  2. Keep the reputation for financial Institutions

3.3 Overall Description

3.3.1 Functional Requirements

Functional user requirements may be high-level statements of what the system should do but functional system requirements should also describe clearly about the system services in detail. The following are the key fields, which should be part of the functional requirements:

- **User :**Execute the task
- **Usability:** This relates to how easily people can use your app. A measure of usability could be the time it takes for end users to become familiar with your app’s functions, without training or help.
- **Reliability:** This is the percentage of time that your app works correctly to deliver the desired results, despite potential failures in its environment.
- **Performance:** This is essentially how fast your app works. A performance requirement for the app could be start in less than 20 seconds.
- **Responsiveness:** This requirement ensures that your app is ready to respond to a user’s input or an external event no matter what it’s doing currently

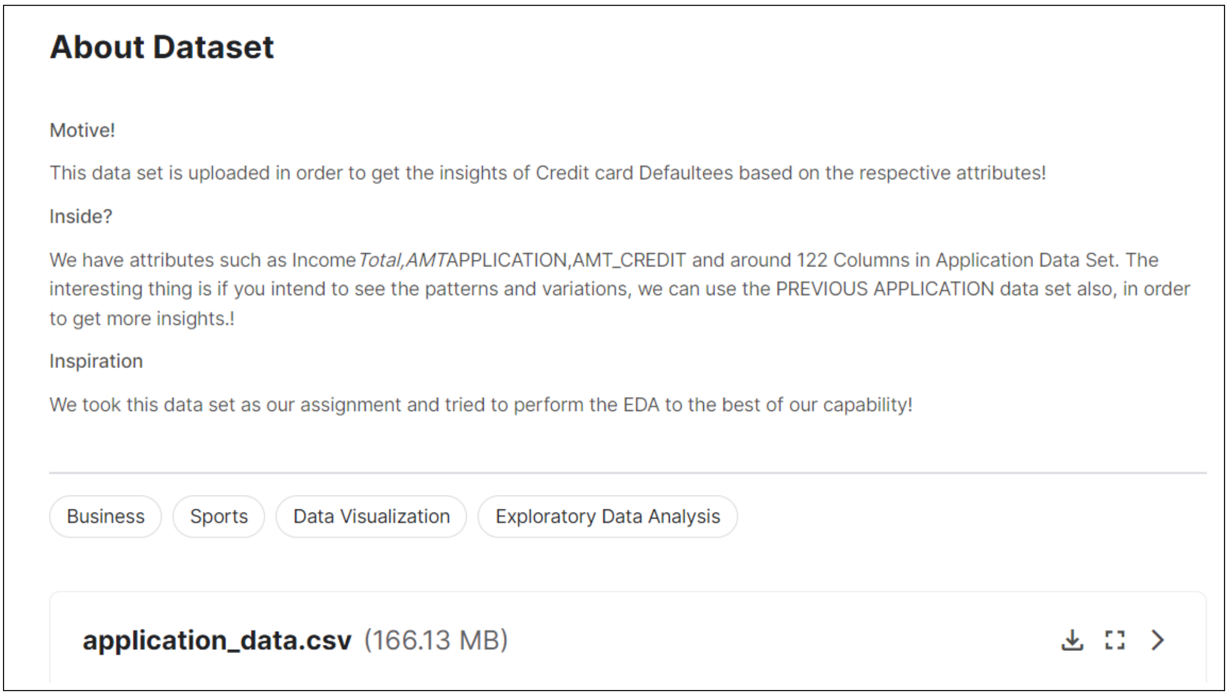
3.3.2 Non-Functional Requirements

- Execution Qualities usability which is observable during operation(at run-time).
- Evolution qualities such as maintainability, extensibility and scalability, which are embodied in the static structure of the system.

3.4 System Requirement

3.4.1 Database Requirement

1. Dataset: Credit Card Fraud Detection



About Dataset

Motive!

This data set is uploaded in order to get the insights of Credit card Defaultees based on the respective attributes!

Inside?

We have attributes such as Income *Total,AMT*APPLICATION,AMT_CREDIT and around 122 Columns in Application Data Set. The interesting thing is if you intend to see the patterns and variations, we can use the PREVIOUS APPLICATION data set also, in order to get more insights.!

Inspiration

We took this data set as our assignment and tried to perform the EDA to the best of our capability!

application_data.csv (166.13 MB)

Figure 3.1: Dataset

3.4.2 Software Requirement

1. Technology Used : Python

2. IDE: Python IDE

3. Operating System : Windows XP or above

3.4.3 Hardware Requirement

1. Hard Disk : 256 GB and above

2. RAM: 8 GB

3. Processor : Intel i5

3.5 Software Engineering Methodology

3.5.1 Software Life-cycle used in this Project

The Waterfall Model was first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed fully before the next phase can begin. This type of model is basically used for the for the project which is small and there are no uncertain requirements. At the end of each phase, a review takes place to determine if the project is on the right path and whether or not to continue or discard the project. In this model the testing starts only after the development is complete. In waterfall model phases do not overlap.

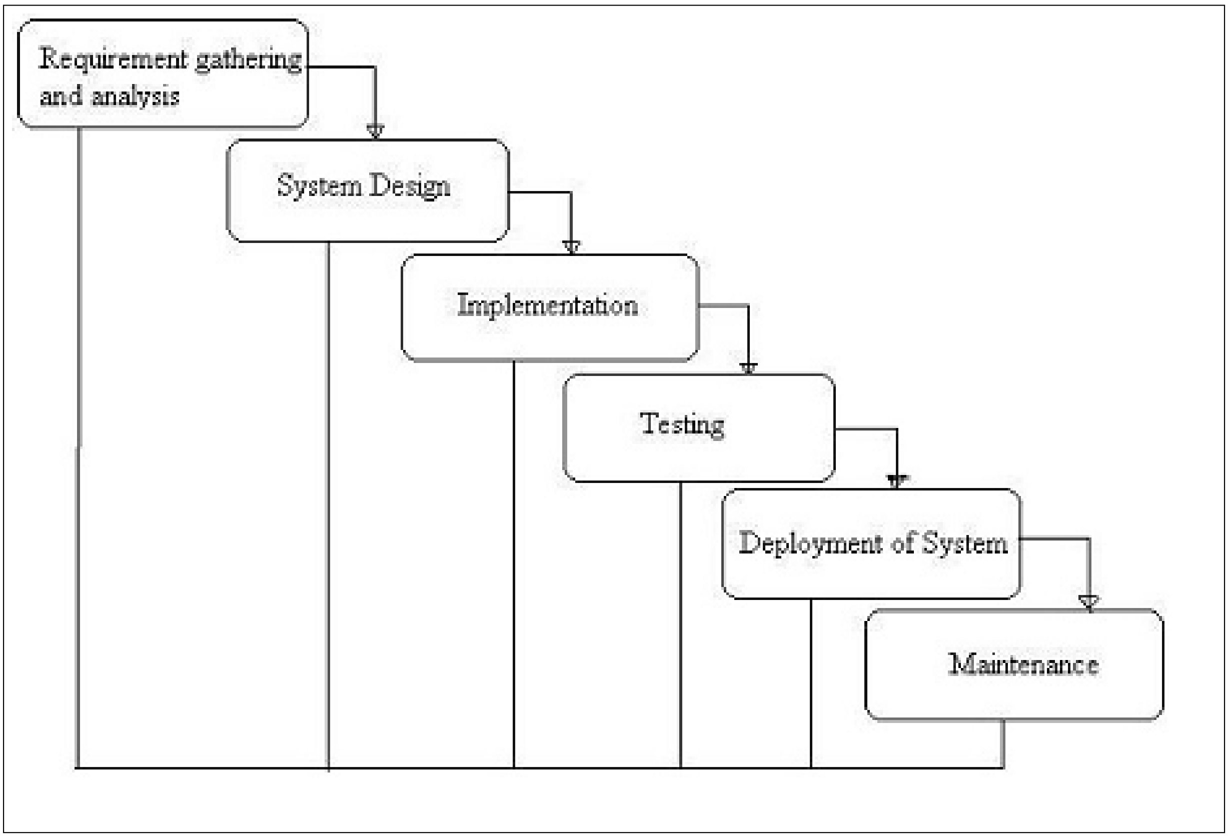


Figure 3.2: Waterfall Model

Chapter 4

Detailed Design

4.1 System Architecture

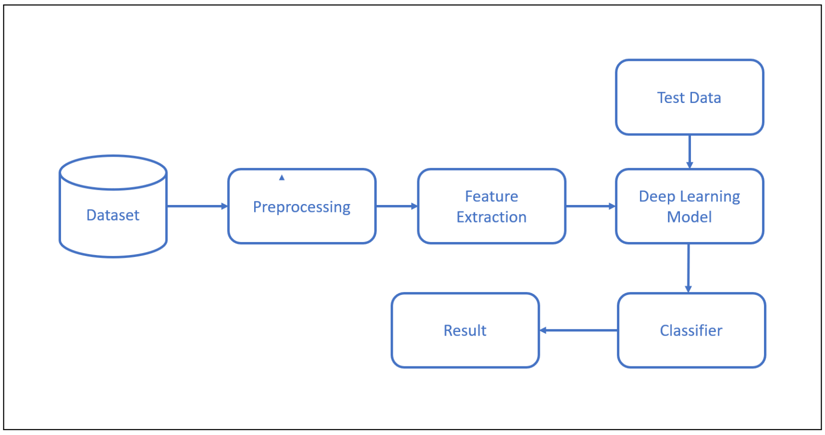


Figure 4.1: Architecture diagram

The diagram shows the major components of the system, which are:

- Dataset & Preprocessing: The ready dataset is taken for credit card fraud detection.A technique which is used to transform the raw data in a useful and efficient format.
- Feature Extraction : The required features are extracted to train the model.
- Model & Classifier : The model is trained with data and classified.
- Test Data: After the model is built, testing data once again validates that it can make accurate predictions. If training and validation data include labels to monitor performance metrics of the model, the testing data should be unlabeled. Test data provides a final, real-world check of an unseen dataset to confirm that the ML algorithm was trained effectively.

4.2 Data Flow Diagram

A data flow diagram (DFD) is a graphical representation of the “flow” of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing.

4.2.1 Level 0 Data Flow Diagram

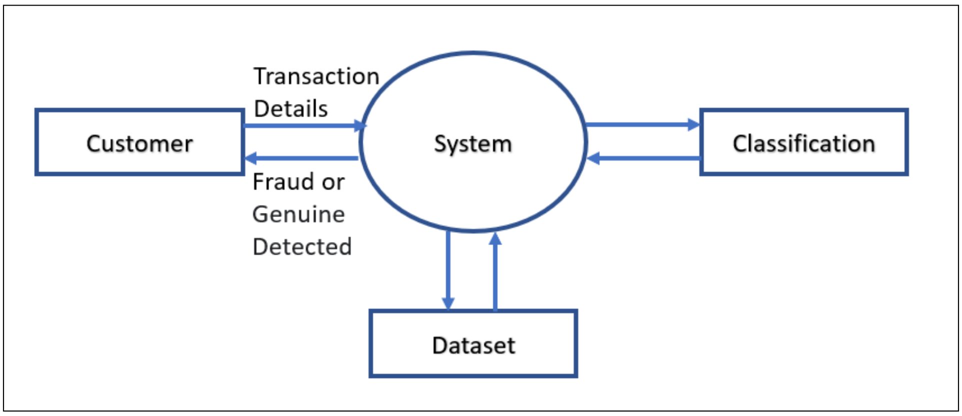


Figure 4.2: Level 0 Data Flow Diagram

4.2.2 Level 1 Data Flow Diagram

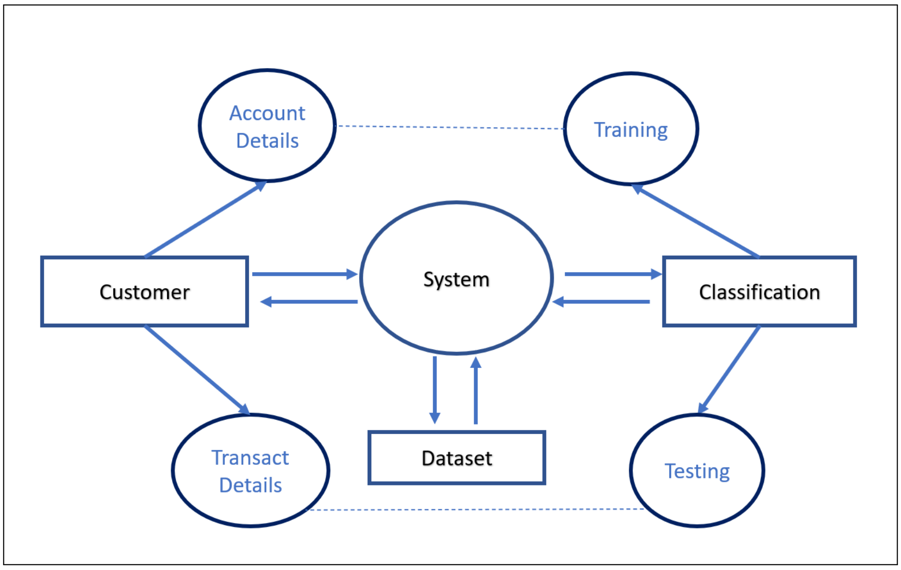


Figure 4.3: Level 1 Data Flow Diagram

4.3 UML

4.3.1 Use-case Diagram

Use case diagrams are typically used to model circumstances where your system or application interfaces with outside entities, which are known as actors. Use case diagrams are usually referred to as behavior diagrams used to describe a set of actions (use cases) that some system or systems (subject) should or can perform in collaboration with one or more external users of the system (actors).

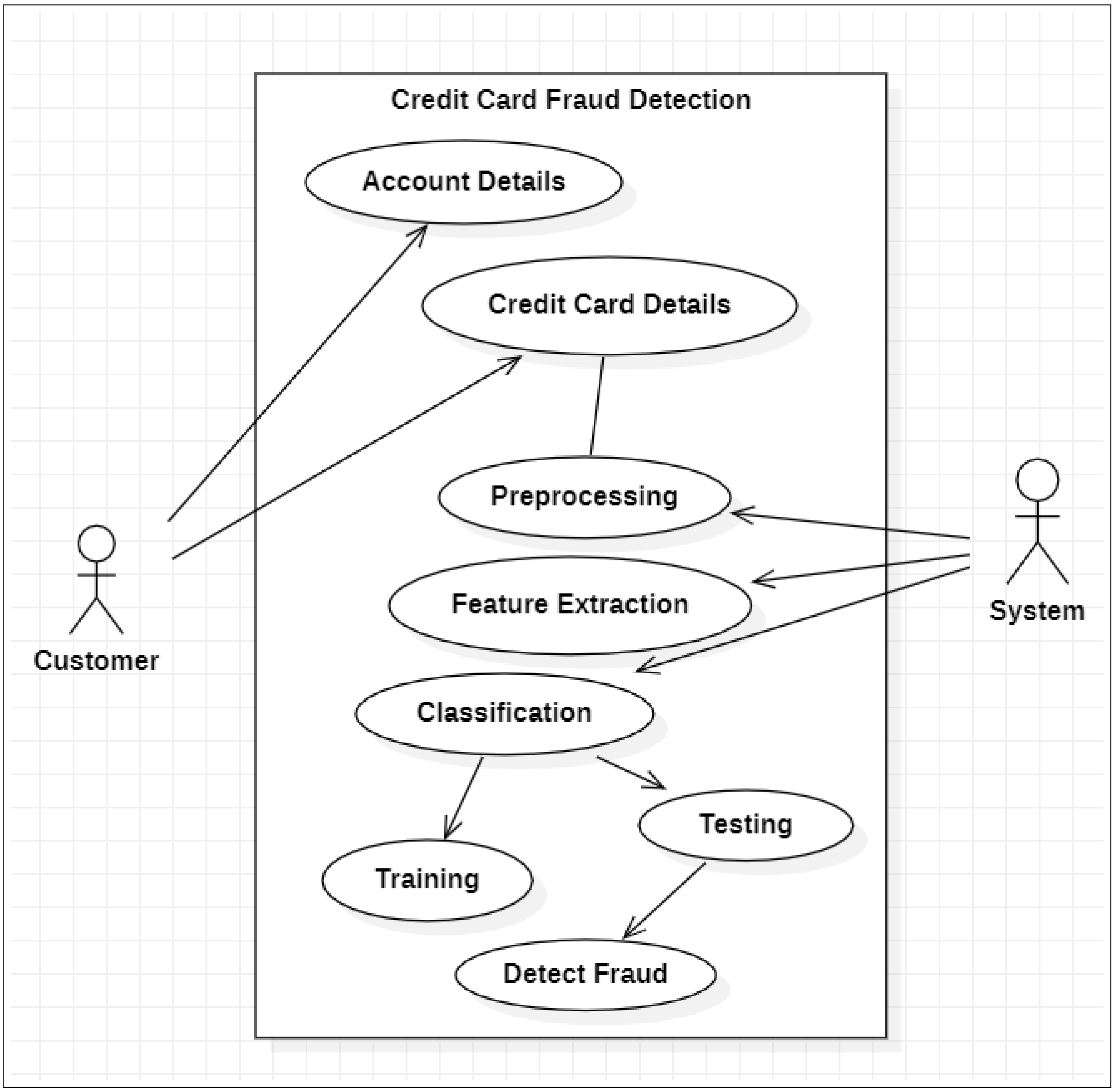


Figure 4.4: Use case diagram

4.3.2 Class Diagram

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the systems classes, their attributes, operations (or methods), and the relationships among objects.

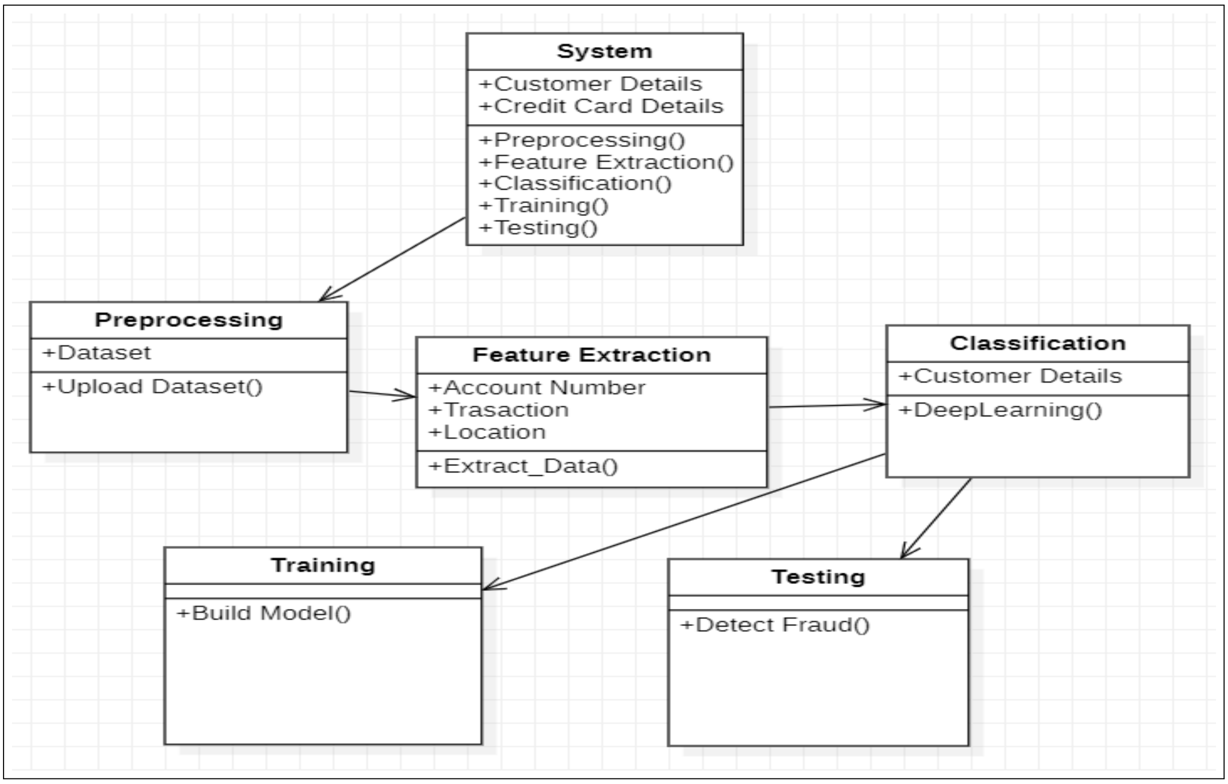


Figure 4.5: Class Diagram

4.3.3 Sequence Diagram

Sequence diagrams provide a graphical representation of object interactions over time. These typically show a user or actor, and the objects and components they interact with in the execution of a use case. One sequence diagram typically represents a single Use Case ’scenario’ or own of events. Sequence diagrams are an excellent way of documenting usage scenarios and both capturing required objects early in analysis and verifying object use later in design. The diagrams show the own of messages from one object to another, and as such correspond to the methods and events supported by a class/object.

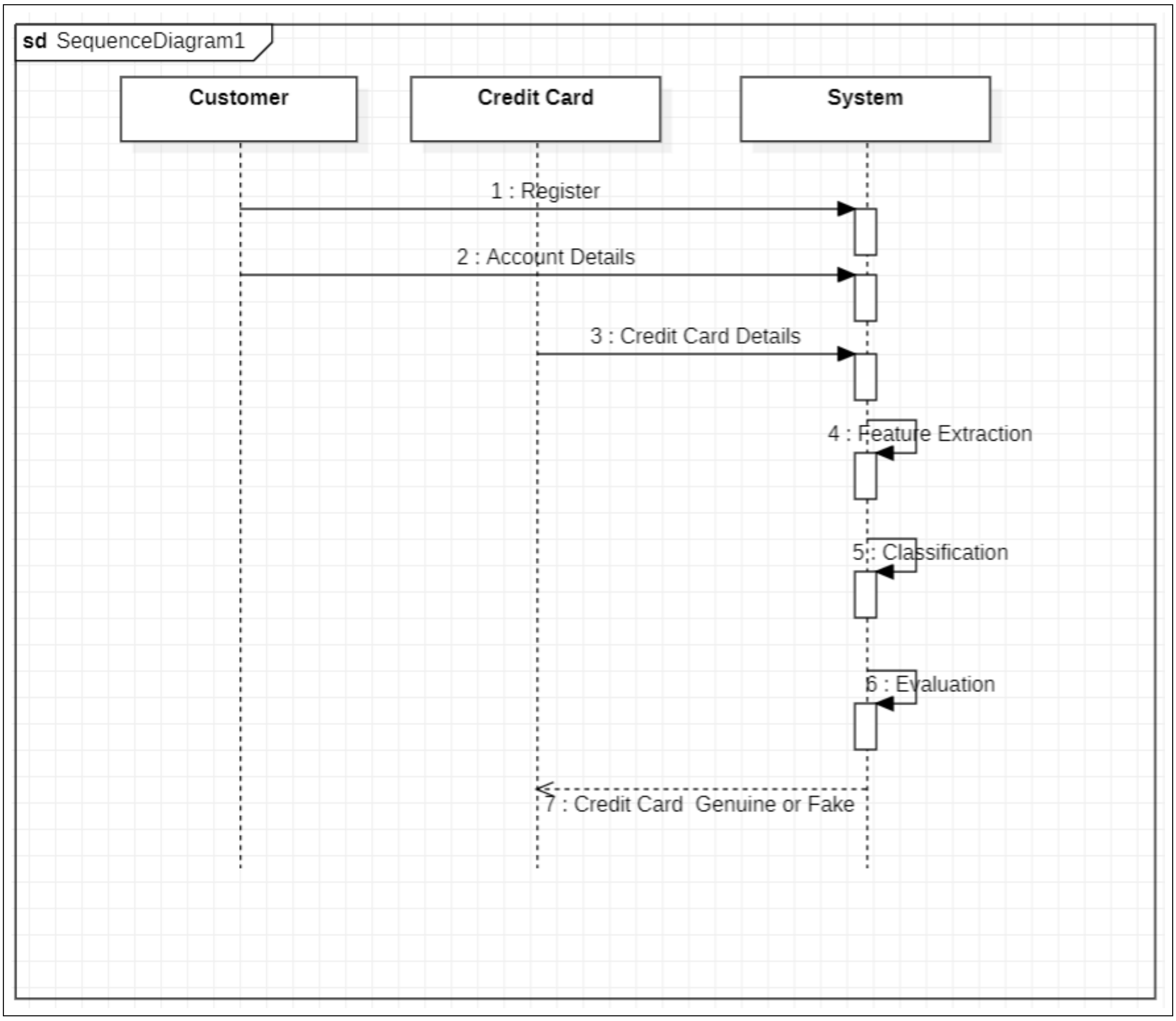


Figure 4.6: Sequence Diagram

4.3.4 Activity Diagram

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

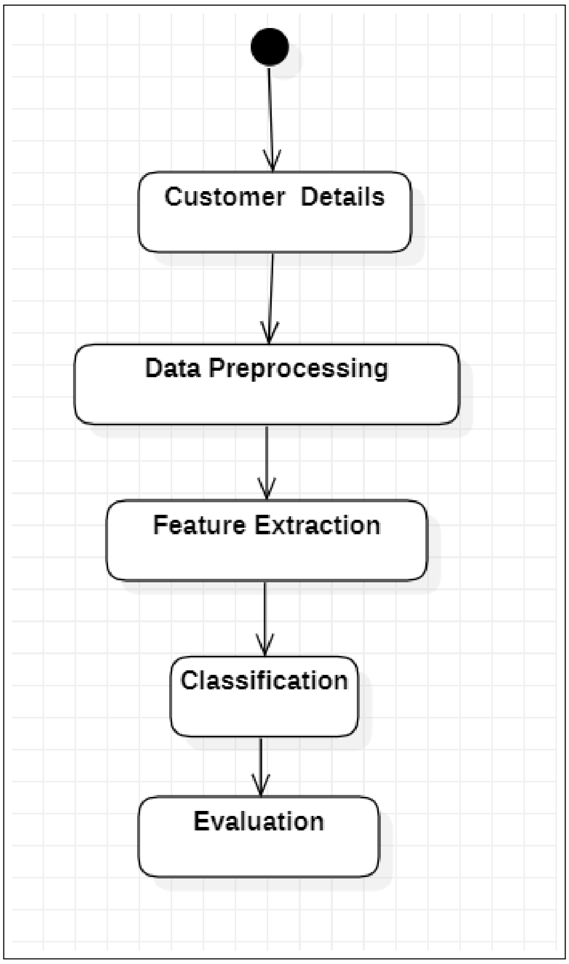


Figure 4.7: Activity Diagram

Chapter 5

Implementation

5.1 Methodology

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. As technology changes, it becomes difficult to track the Modeling and pattern of fraudulent transactions. With the rise of machine learning, artificial intelligence and other relevant fields of information technology, it becomes feasible to automate this process and to save some of the intensive amount of labour that is put into detecting credit card fraud,

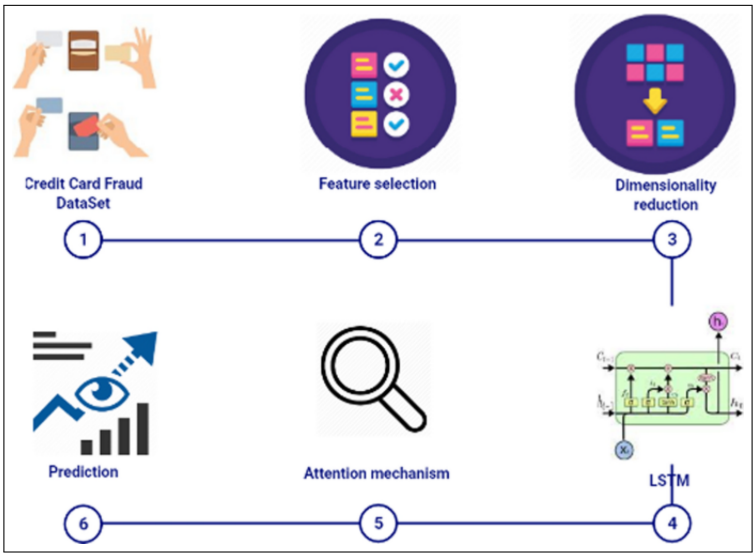


Figure 5.1: Flow

5.2 Algorithms

5.2.1 Logistic Regression Algorithm

Logistic regression algorithm, sometimes called logit model, is a common model for dichotomous output variables and was extended for disease classification prediction. Suppose that there are p input variables where their values are indicated by x_1, x_2, ..., x_p. Let z be a probability that an event will occur and 1-z be a probability that the event will not occur. The logistic regression model is given by

$$log(\frac{z}{1-z}) = logit(z) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p$$

or can be written by

$$\text{Z} = \frac{e^{(\beta_0+\beta_1 x_1+\beta_2 x_2+...+\beta_p x_p)}}{1+e^{(\beta_0+\beta_1 x_1+\beta_2 x_2+...+\beta_p x_p}}$$

where 0 is the intercept and β_0, β_1, ..., β_p are the regression coefficients.

Chapter 6

Results and Evaluation

6.1 Experimental Setup

6.2 Testing Strategy

Testing is a vital part of software development, and it is important to start it as early as possible, and to make testing a part of the process of deciding requirements. Testing is part of a lifecycle. The software development lifecycle is one in which we hear of a need, we write some code to fulfill it, and then we check to see whether you we pleased the stakeholders—the users, owners, and other people who have an interest in what the software does.

We therefore design tests based on the stakeholders' needs, and run the tests before the product reaches the users. Preferably well before then, so as not to waste our time working on something that isn't going to do the job.

- Tests represent requirements. Whether you write user stories on sticky notes on the wall, or use cases in a big thick document, your tests should be derived from and linked to those requirements. And as we've said, devising tests is a good vehicle for discussing the requirements.

- We're not done till the tests pass. The only useful measure of completion is when tests have been performed successfully.

Test Objective:

The objective of our test plan is to find and report as many bugs as possible to improve the integrity of our program. Although exhaustive testing is not possible, we will exercise a broad range of tests to achieve our goal.

Following test process approach will be followed :

- Organize Project involves creating a System Test Plan, Schedule and Test Approach, and assigning responsibilities.
- Design/Build System Test involves identifying Test Cycles, Test Cases, Entrance and Exit Criteria, Expected Results, etc. In general, test conditions/expected results will be identified by the Test Team in conjunction with the Development Team. The Test Team will then identify Test Cases and the Data required. The Test conditions are derived from the Program Specifications Document.
- Design/Build Test Procedures includes setting up procedures such as Error Management systems and Status reporting.
- Build Test Environment includes requesting/building hardware, software and data set-ups.
- Execute System Tests – The tests identified in the Design/Build Test Procedures will be executed. All results will be documented and Bug Report Forms filled out and given to the Development Team as necessary.
- Signoff - Signoff happens when all pre-defined exit criteria have been achieved.

UNIT TESTING:

During the unit testing phase, the system is tested while it is developed. Here all the options of the system are validated.

During the first phase of this testing the testing person tests the system by entering the valid data, or by performing the appropriate function which the system requests for. This phase of testing is done to verify whether the system performs all the requested functions.

White Box Testing:

White box testing, sometimes called as glass box testing, is a test case design method that uses control structure of the procedural design to derive the test cases. Using the white box testing methods, the software engineer can derive the test case that:

- Guarantee that all independent paths within module having exercised at least once.
- Exercise all logical decisions and their true or false sides.
- Execute all loops and within their operational bounds.
- Exercise the internal data structures to ensure their validity.

Black Box Testing:

We have focused on the Functional requirement and the proceeding of the module to fulfil the requirement the requirement will be appreciated by the module instances and give the desired functionality as per the expectations. Black box testing attempts to find the errors in following categories:

- Incorrect user details
- Interface problems
- Initialization Complexities
- Performance Measurement
- Error in data structures and external database error

Integration Testing:

The integration will be based on the total whole module testing the integration of the module will be categories as follows:

- Checking the module and Data entry status
- Redirectional functionalities
- Checking the Status through ID

- Checking Chart Accuracy
- Error in data structures and external database error

Top Down Integration:

Top down integration is an incremental approach to the construction of program structure modules and integrated by moving downwards through the control hierarchy beginning with main control module. Modules subordinates to the main control module are incorporated into the structure in either the depth first or breadth first manner as per the categories:

- Sending data to the database
- Constraint validation
- Updating the further changes in the system
- Looking for the Status
- Perform the Graphical Accuracy Test

.

Chapter 7

Conclusion

The main objective of the system is the identification of credit card fraud. This system proposes a method which not only involves simple construction with associated attributes but is with better algorithms to detect Fraud. The proposed application will improve the level of awareness and identification of the fake or counterfeit in circulation to the people who used cash transactions regularly in their various businesses

References

[1] Y. Singh, K. Singh and V. Singh Chauhan, ”Fraud Detection Techniques for Credit Card Transactions,” 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), 2022, pp. 821-824, doi: 10.1109/ICIEM54221.2022.9853183.

[2] S. K. Saddam Hussain, E. Sai Charan Reddy, K. G. Akshay and T. Akanksha, ”Fraud Detection in Credit Card Transactions Using SVM and Random Forest Algorithms,” 2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2021, pp. 1013-1017, doi: 10.1109/I-SMAC52330.2021.9640631.

[3] A. K. Rai and R. K. Dwivedi, ”Fraud Detection in Credit Card Data using Unsupervised Machine Learning Based Scheme,” 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), 2020, pp. 421-426, doi: 10.1109/ICESC48915.2020.9155615.

[4] V. Jain, M. Agrawal and A. Kumar, ”Performance Analysis of Machine Learning Algorithms in Credit Cards Fraud Detection,” 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), 2020, pp. 86-88, doi: 10.1109/ICRITO48877.2020.9197762.

[5] F. K. Alarfaj, I. Malik, H. U. Khan, N. Almusallam, M. Ramzan and M. Ahmed, ”Credit Card Fraud Detection Using State-of-the-Art Machine Learning and Deep Learning Algorithms,” in IEEE Access, vol. 10, pp. 39700-39715, 2022, doi: 10.1109/ACCESS.2022.3166891.(BASE Paper)