GUJARAT TECHNOLOGY UNIVERSITY

Government Engineering College-Bhavnagar

A

Report

On

Financial Firewall: Credit Card Security AI

Under subject of

Summer Internship (3170001)

B.E. Semester–VII

(Computer Engineering)

Submitted by:

Kheni Tapan Shambhubhai (210210107043)

Under the supervision of

Internal Guide: **Prof. Vishal Andodariya**

Prof. K.P. Kandoriya

Head of the Department

Computer Engineering Department

Academic year

(2024-25)

Government Engineering college, Bhavnagar

Nr. Sir BPTI Campus, Vidhyanagar, Bhavnagar, Gujarat 364002

DECLARATION

I hereby declare that the Project report submitted along with the Project entitled ‘Financial Firewall: Credit Card Security AI’ submitted in partial fulfilment for the subject of summer internship (3170001) for degree of Bachelor of Engineering in computer engineering to Gujarat Technological University, Ahmedabad, is a bonafide record of original project work carried out by me at Government Engineering College, Bhavnagar under the supervision of **Prof. Vishal Andodariya** and that no part of this report has been directly copied from any students’ reports or taken from any other source, without providing due reference.

Name of student: Kheni Tapan Shambhubhai

Enrolment Number: 210210107043

Signature of student:

CERTIFICATE

This is to certify that the report entitled **“**Financial Firewall: Credit Card Security AI” has been carried out by **Kheni Tapan Shambhubhai (210210107043)** Under my guidance in fulfilment of the subject Summer Internship (3170001) of Bachelor of Engineering in COMPUTER ENGINEERING (7th Semester) of Gujarat technological University, Ahmedabad during the academic year 2024-25.

Internal Guide	Head of Department
Prof. Vishal Andodariya	Prof. K.P. Kandoriya
Assistant Professor	Head of Department
Computer Engineering	Computer Engineering Department
GEC, Bhavnagar	GEC, Bhavnagar

ACKNOWLEDGEMENT

We are having great pleasure in submitting this project report on “Financial Firewall: Credit Card Security AI” as part of the curriculum of Summer Internship (3170001). I convey thanks to my internal guide **Prof. Vishal Andodariya**, Computer Engineering Department, Government Engineering College – Bhavnagar for providing encouragement, constant support and. I am grateful to **Prof. (Dr.) M. G. Bhatt**, Principal of Government Engineering College – Bhavnagar for giving us the support and encouragement that was necessary for the completion of this internship.

Kheni Tapan Shambhubhai

210210107043

ABSTRACT

This project develops a predictive model for assessing credit card default and fraud risk using advanced machine learning techniques. Leveraging comprehensive credit card holder and transaction datasets, we implement and compare multiple algorithms, including logistic regression, random forests, and gradient boosting machines.

The methodology addresses class imbalance issues common in financial data through advanced sampling and parameter tuning.

Our results provide high-performing predictive models and identify key risk indicators, offering valuable insights for financial institutions to enhance their credit risk assessment and fraud detection processes.

The analysis of feature importance provides a clear understanding of the most significant factors contributing to credit card defaults and fraudulent transactions.

Technologies used in This Project:

***Language**: Python*

***Framework**: Scikit-learn, Streamlit*

***Database**: MongoDB*

List of Figure

TABLE OF CONTENTS

1. INTRODUCTION

1.1 PROJECT OVERVIEW

The **Financial Firewall: Credit Card Security AI** is a machine learning-based system designed to enhance the security of financial institutions by preventing fraudulent credit card transactions and identifying potential customers who are at risk of defaulting on their payments. The project leverages two datasets—one for fraudulent transaction detection and the other for customer default prediction. The system has been built to achieve high accuracy in both areas, with multiple machine learning models trained and optimized to ensure reliable and robust predictions.

In addition to the core machine learning models, a **Streamlit-based web application** has been developed. This application allows users to interact with the models, train them on new data, and make predictions in real-time. The application is user-friendly, providing multiple pages for different functions such as model training, prediction, and basic information about the system and its developers.

1.2 SCOPE

The scope of the Financial Firewall AI includes:

1. **Fraudulent Transaction Detection:**
    - Identify and prevent fraudulent credit card transactions by analyzing transaction data in real-time.
    - Help financial institutions reduce losses from fraudulent activities and protect customer accounts.

2. **Customer Default Prediction:**
    - Predict which customers are likely to default on their payments based on historical and behavioral data.

- Enable banks and financial services to manage credit risks better and make informed lending decisions.

3. **Web Application for Interactivity:**
   - A platform for users (e.g., data scientists or financial analysts) to train models with new data.
   - Users can input transaction or customer data to receive real-time predictions for fraud detection or default risk assessment.

4. **Potential Future Enhancements:**
   - Adding more datasets to improve model generalization.
   - Integrating advanced algorithms such as deep learning to enhance performance further.
   - Scaling the system to handle larger datasets and real-time data processing.

1.3 OBJECTIVE

Fraud Prevention:

Detect fraudulent transactions with high accuracy to protect users and financial institutions from losses.

Credit Risk Assessment:

Predict customer payment default risks to help lenders manage their loan portfolios more effectively.

High Model Accuracy:

Develop models that achieve a minimum of 99% accuracy in both fraudulent transaction detection and default prediction.

User-Friendly Application:

Create a simple, yet powerful interface using Streamlit that allows users to train models and make predictions with ease.

Flexibility and Adaptability:

Build a flexible system where new data can be introduced, models can be retrained, and predictions can be made dynamically.

2. SYSTEM ANALYSIS

2.1 USER CHARACTERISTICS

1. **Financial Analysts:**
   Users with domain knowledge in finance and credit risk management but limited expertise in machine learning. They can use the prediction functionalities to assess transaction fraud risk or customer default risk, providing them with actionable insights to inform decisions on credit policies, fraud detection, and resource allocation.

2. **Banking and Financial Institutions:**
   These institutions would be the primary users of the system, employing it as a tool to safeguard against credit card fraud and optimize lending operations by assessing customer risk. Security and accuracy are critical for these users, and the web interface allows them to deploy and use the models without requiring deep technical knowledge.

3. **Developers/Technologists:**
   Users with expertise in software development and AI implementation. They will be able to modify or enhance the current system, integrate additional features, and deploy the solution in a production environment for wider use.

2.2 TOOLS & TECHNOLOGY

The project makes use of a variety of tools and technologies to ensure robustness, efficiency, and accessibility. These include:

1. **Programming Language:**
   - **Python:** The project is developed in Python, which offers rich libraries for machine learning, data manipulation, and web development.

2. **Machine Learning Libraries:**
   - **Scikit-learn:** Used for training and implementing various machine learning models like Logistic Regression, Decision Trees, Random Forest,

- and more for both fraud detection and default prediction.
- **XGBoost/LightGBM:** Potentially used for improving model accuracy and performance, especially in handling large datasets.
- **Pandas & NumPy:** For data preprocessing, handling missing values, and performing operations on large datasets efficiently.

3. **Data Visualization:**
   - **Matplotlib and Seaborn:** For generating insightful graphs and charts during the exploratory data analysis (EDA) phase.

4. **Web Framework:**
   - **Streamlit:** Used for developing the user-friendly web application that allows interaction with machine learning models, including training and prediction pages.

5. **Version Control:**
   - **Git/GitHub:** For managing code, versioning, and collaboration in the development of the project.

2.3 NOVELTY IN PROJECT WORK

1. **Dual-Purpose System:**
   The project addresses two critical financial challenges—fraud detection and customer default prediction—under a single framework. Most projects in the financial space focus on one aspect, but this system integrates both, making it a comprehensive solution for financial institutions.

2. **Streamlined User Interface with Streamlit:**
   While many machine learning models are developed for offline use, this project leverages Streamlit to create an interactive, web-based interface that makes model training and predictions accessible to non-technical users. This focus on user-friendliness without compromising on the technical sophistication of the

underlying models is a key innovation.

3. **High Accuracy of Predictions:**
   Achieving around 99% accuracy in both fraud detection and default prediction is a notable achievement. The project employs advanced machine learning models, rigorous feature engineering, and optimization techniques to deliver high performance, surpassing many standard implementations.

4. **Customizability and Flexibility:**
   The ability for users to switch between different prediction tasks (fraud or default) and input data for live predictions demonstrates flexibility, making the system adaptable for various financial contexts.

5. **End-to-End Solution:**
   From data preprocessing, model training, and evaluation, to deployment and real-time predictions, the project covers the entire machine learning lifecycle. This full-spectrum approach distinguishes the project as a ready-to-use solution rather than a conceptual model.

3. IMPLEMENTATION ENVIRONMENT

3.1 IMPLEMENTATION ENVIRONMENT

1. **Single vs Multi-user Environment:**

   The current version of the web application is primarily designed for single-user interaction. The system allows one user at a time to train models, input data, and receive predictions. This setup is suitable for data scientists, analysts, or financial institutions using the application locally or on a server for their personal or team-specific use.

   While the project is initially set up for single-user operation, it can be deployed on a server (e.g., via Streamlit Cloud, AWS, or Heroku) to support multi-user access. This would enable multiple users from different locations to interact with the system simultaneously, as the web app and model handling can support concurrent sessions.

2. **GUI vs Non-GUI**:

   The project is implemented with a **graphical user interface (GUI)** using **Streamlit**. This makes the system highly user-friendly and easy to navigate for users with minimal technical expertise. Features such as drop-downs, buttons, and data input fields allow users to interact with the model seamlessly.

   For advanced users or developers who prefer working in a **non-GUI** environment (e.g., via command line or automated scripts), the backend of the project is written in **Python**. The core machine learning functionality can be accessed by executing scripts directly or via API endpoints if the system is extended.

3.2 CODING STANDARDS

The project follows common Python coding standards and best practices, including:

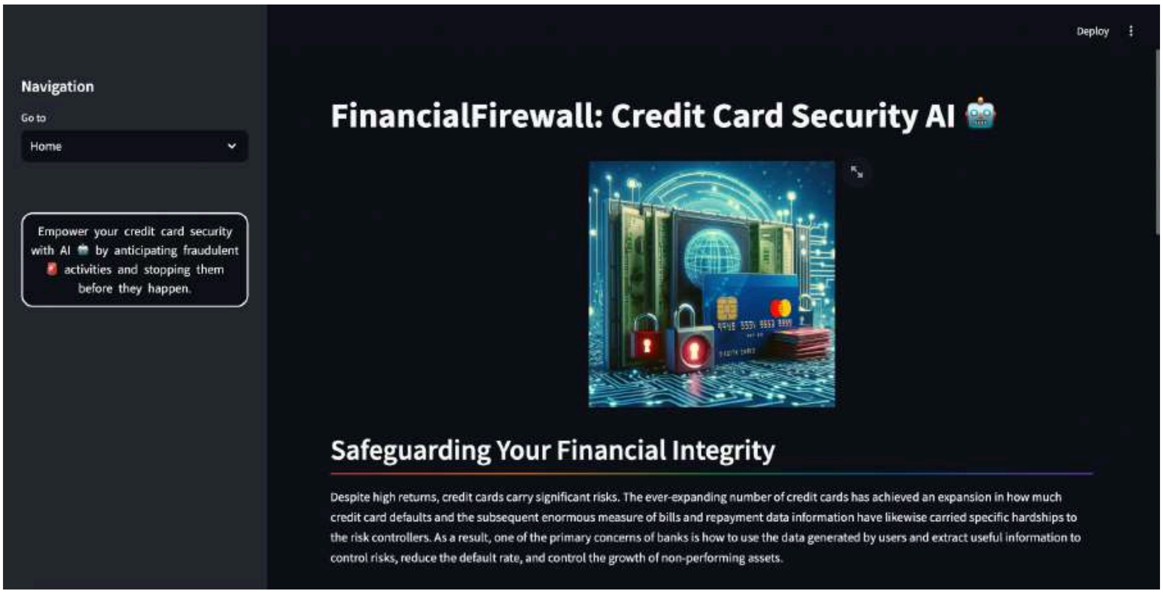
- PEP 8 guidelines for readability and style.
- Modular code structure for reusability and separation of concerns.
- Docstrings for functions and classes to explain their functionality.

Sample code of the main.py file:

```
from CreditCardFraudDetection.pipeline.stage_01_data_ingestion import
DataIngestionPipeline
from CreditCardFraudDetection.pipeline.stage_02_data_validation import
DataValidationPipeline
from CreditCardFraudDetection.pipeline.stage_03_data_transformation import
DataTransformationPipeline
from CreditCardFraudDetection.pipeline.stage_04_model_trainer import
ModelTrainerPipeline
from CreditCardFraudDetection.pipeline.stage_05_model_evaluation import
ModelEvaluationPipeline

from CreditCardFraudDetection import logger

STAGE_NAME = "data ingestion stage"

try:
    logger.info(f">>>>>>> stage {STAGE_NAME} started <<<<<<<")
    obj = DataIngestionPipeline()
    obj.main()
    logger.info(f">>>>>>> stage {STAGE_NAME} completed
<<<<<<<\n\nx==========x")
except Exception as e:
    logger.exception(e)
    raise e

STAGE_NAME = "data validation stage"

try:
    logger.info(f">>>>>>> stage {STAGE_NAME} started <<<<<<<")
    obj = DataValidationPipeline()
    obj.main()
    logger.info(f">>>>>>> stage {STAGE_NAME} completed
<<<<<<<\n\nx==========x")
except Exception as e:
    logger.exception(e)
    raise e
```

```
STAGE_NAME = "data transformation stage"

try:
   logger.info(f">>>>>>> stage {STAGE_NAME} started <<<<<<<")
   obj = DataTransformationPipeline()
   obj.main()
   logger.info(f">>>>>>> stage {STAGE_NAME} completed
<<<<<<<\n\nx==========x")
except Exception as e:
   logger.exception(e)
   raise e

STAGE_NAME = "model trainer stage"

try:
   logger.info(f">>>>>>> stage {STAGE_NAME} started <<<<<<<")
   obj = ModelTrainerPipeline()
   obj.main()
   logger.info(f">>>>>>> stage {STAGE_NAME} completed
<<<<<<<\n\nx==========x")
except Exception as e:
   logger.exception(e)
   raise e

STAGE_NAME = "model evaluation stage"

try:
   logger.info(f">>>>>>> stage {STAGE_NAME} started <<<<<<<")
   obj = ModelEvaluationPipeline()
   obj.main()
   logger.info(f">>>>>>> stage {STAGE_NAME} completed
<<<<<<<\n\nx==========x")
except Exception as e:
   logger.exception(e)
   raise e
```

3.3 SNAPSHOTS OF PROJECT

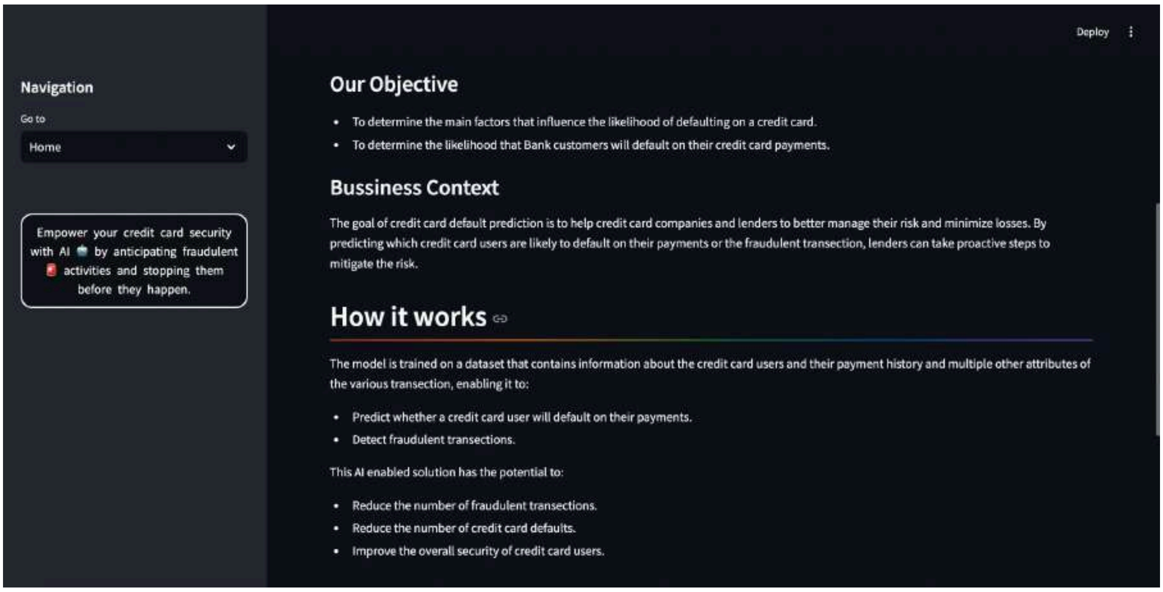


Empower your credit card security with AI 🤖 by anticipating fraudulent 🚨 activities and stopping them before they happen.

FinancialFirewall: Credit Card Security AI 🤖

Safeguarding Your Financial Integrity

Despite high returns, credit cards carry significant risks. The ever-expanding number of credit cards has achieved an expansion in how much credit card defaults and the subsequent enormous measure of bills and repayment data information have likewise carried specific hardships to the risk controllers. As a result, one of the primary concerns of banks is how to use the data generated by users and extract useful information to control risks, reduce the default rate, and control the growth of non-performing assets.



Empower your credit card security with AI 🤖 by anticipating fraudulent 🚨 activities and stopping them before they happen.

Our Objective

- To determine the main factors that influence the likelihood of defaulting on a credit card.
- To determine the likelihood that Bank customers will default on their credit card payments.

Bussiness Context

The goal of credit card default prediction is to help credit card companies and lenders to better manage their risk and minimize losses. By predicting which credit card users are likely to default on their payments or the fraudulent transection, lenders can take proactive steps to mitigate the risk.

How it works

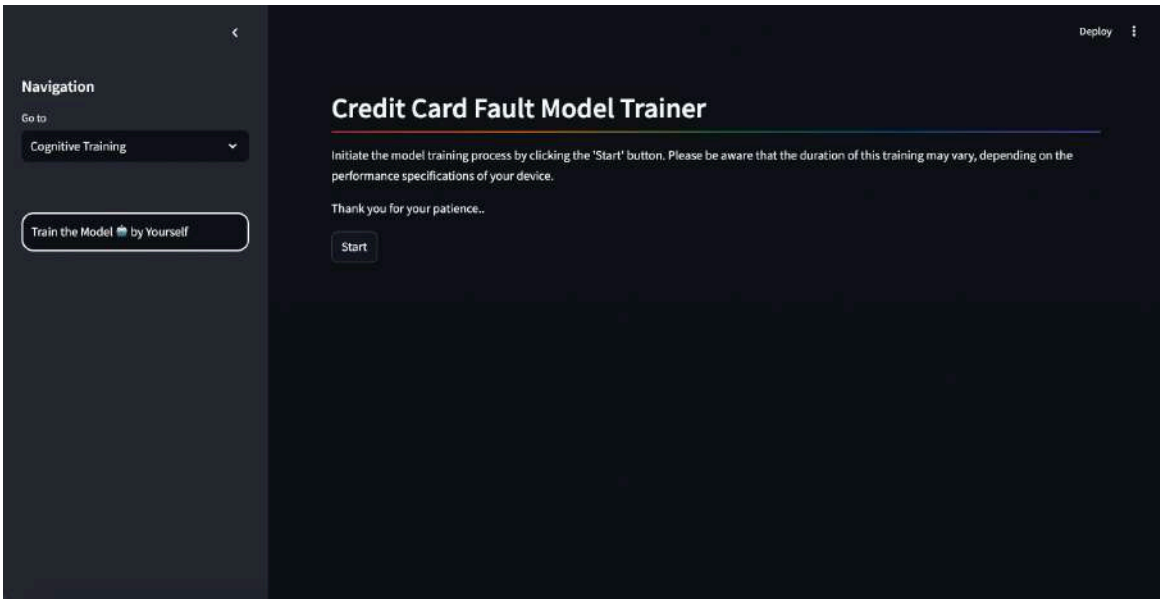
The model is trained on a dataset that contains information about the credit card users and their payment history and multiple other attributes of the various transection, enabling it to:

- Predict whether a credit card user will default on their payments.
- Detect fraudulent transections.

This AI enabled solution has the potential to:

- Reduce the number of fraudulent transections.
- Reduce the number of credit card defaults.
- Improve the overall security of credit card users.

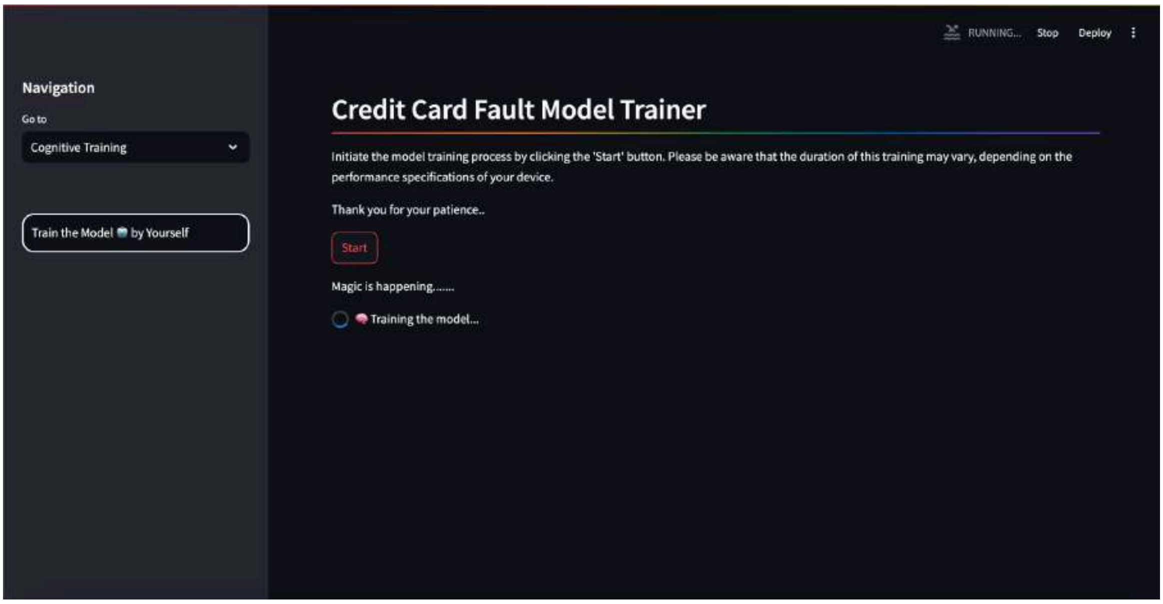
Figure 3.3.1 Home Page



Credit Card Fault Model Trainer

Initiate the model training process by clicking the 'Start' button. Please be aware that the duration of this training may vary, depending on the performance specifications of your device.

Thank you for your patience..



Credit Card Fault Model Trainer

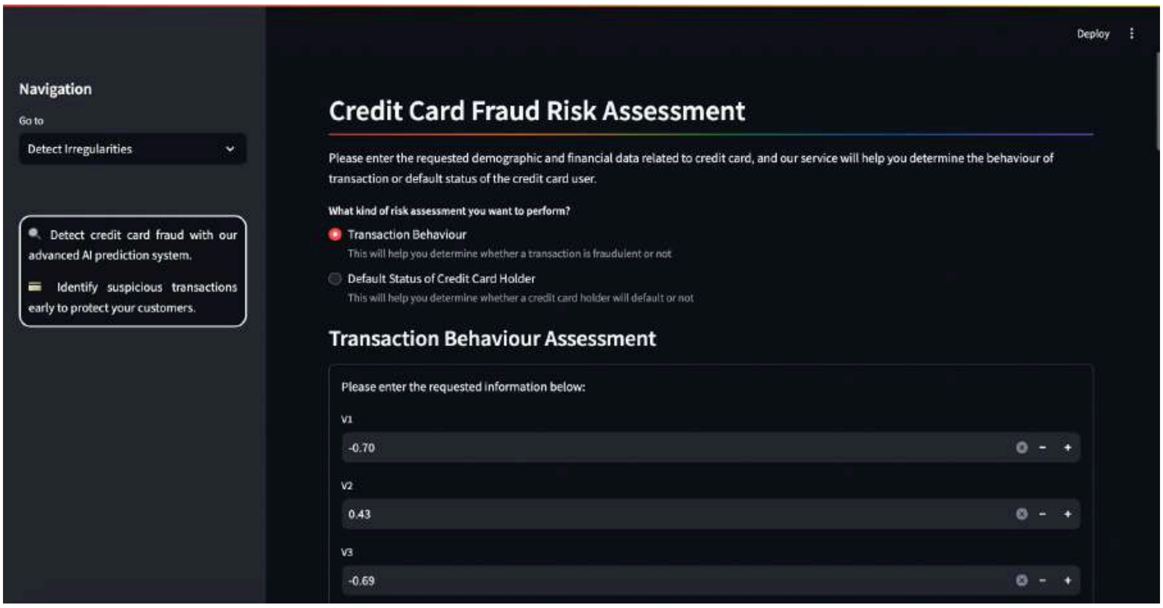
Initiate the model training process by clicking the 'Start' button. Please be aware that the duration of this training may vary, depending on the performance specifications of your device.

Thank you for your patience..

Magic is happening.......

🧠 Training the model...

Figure 3.3.2 Cognitive Training Page



Navigation

Go to

Detect Irregularities

🔍 Detect credit card fraud with our advanced AI prediction system.

💳 Identify suspicious transactions early to protect your customers.

Credit Card Fraud Risk Assessment

Please enter the requested demographic and financial data related to credit card, and our service will help you determine the behaviour of transaction or default status of the credit card user.

What kind of risk assessment you want to perform?

● Transaction Behaviour
This will help you determine whether a transaction is fraudulent or not

○ Default Status of Credit Card Holder
This will help you determine whether a credit card holder will default or not

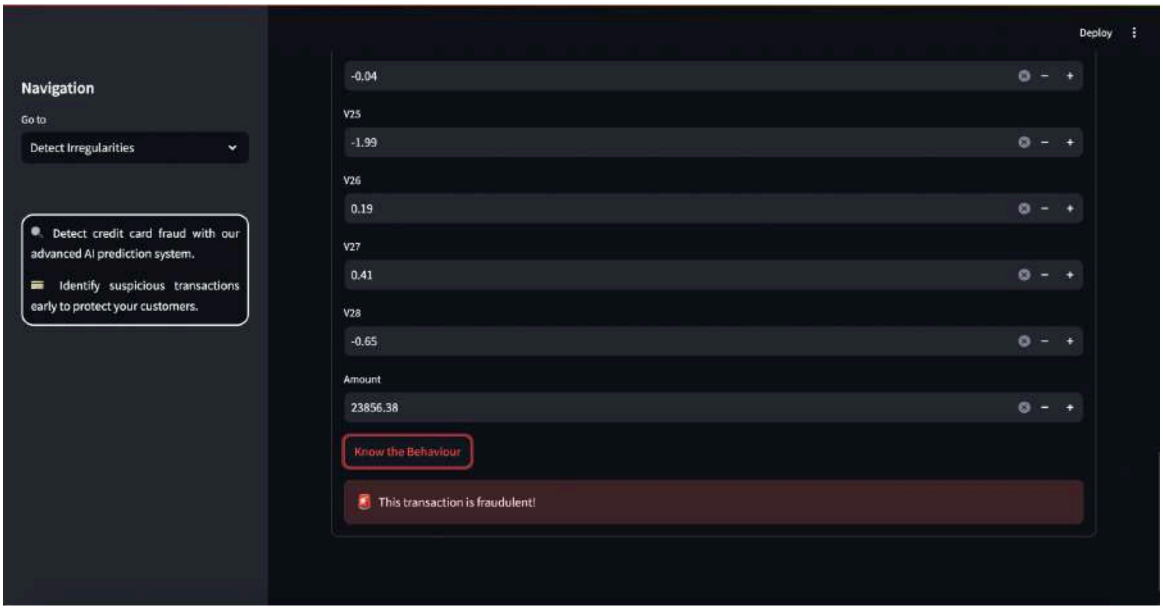
Transaction Behaviour Assessment

Please enter the requested information below:

V1
-0.70

V2
0.43

V3
-0.69



Navigation

Go to

Detect Irregularities

🔍 Detect credit card fraud with our advanced AI prediction system.

💳 Identify suspicious transactions early to protect your customers.

-0.04

V25
-1.99

V26
0.19

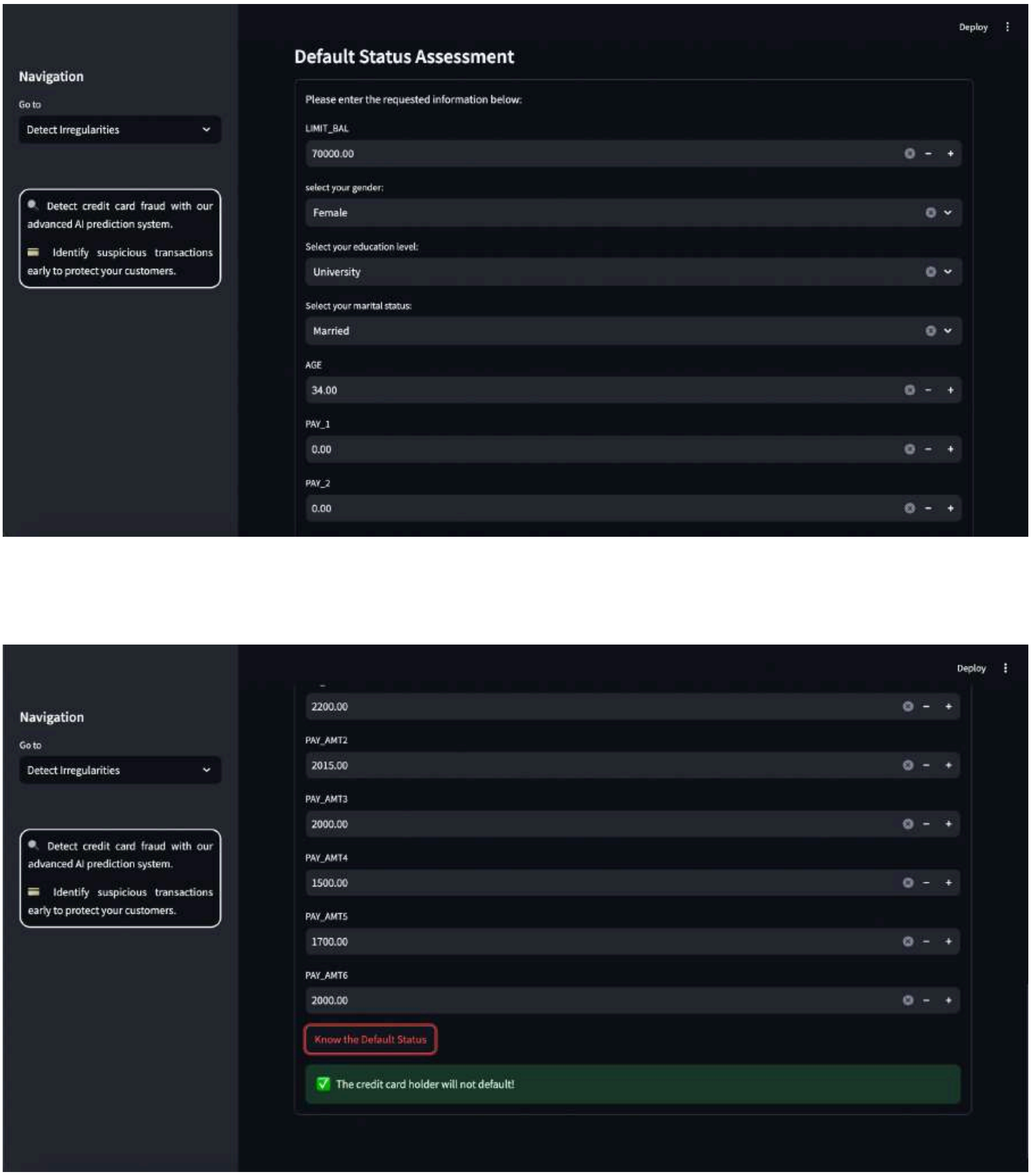
V27
0.41

V28
-0.65

Amount
23856.38

This transaction is fraudulent!

Figure 3.3.3 Prediction Page (Fraudulent Transaction Prediction)



Detect credit card fraud with our advanced AI prediction system.

Identify suspicious transactions early to protect your customers.

Default Status Assessment

Please enter the requested information below:

LIMIT_BAL: 70000.00

select your gender: Female

Select your education level: University

Select your marital status: Married

AGE: 34.00

PAY_1: 0.00

PAY_2: 0.00

Detect credit card fraud with our advanced AI prediction system.

Identify suspicious transactions early to protect your customers.

2200.00

PAY_AMT2: 2015.00

PAY_AMT3: 2000.00

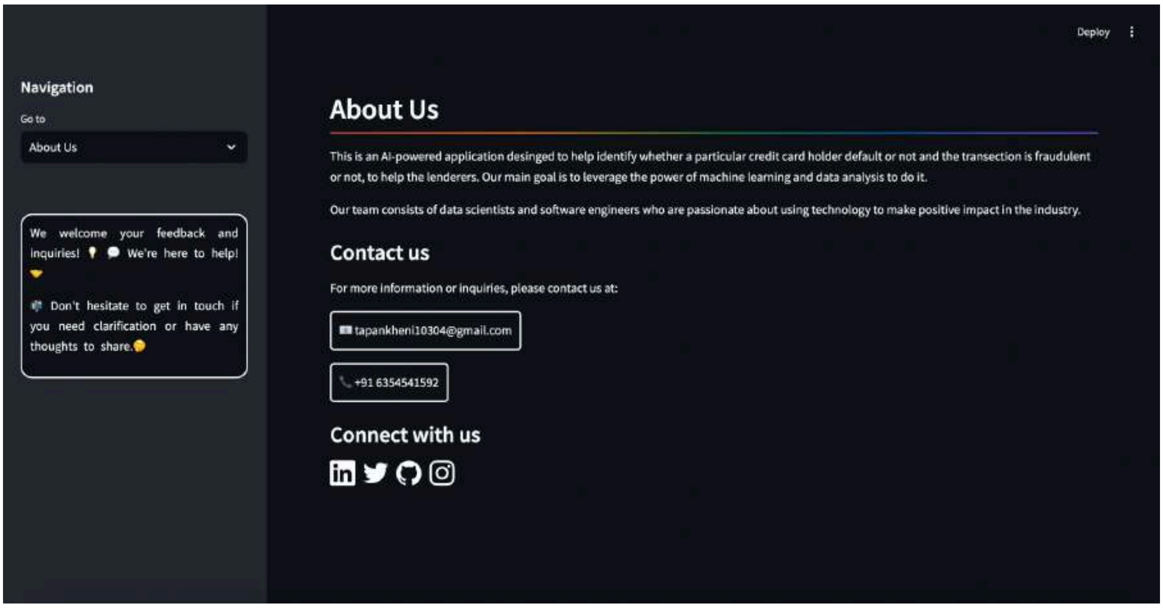
PAY_AMT4: 1500.00

PAY_AMT5: 1700.00

PAY_AMT6: 2000.00

✅ The credit card holder will not default!

Figure 3.3.4 Prediction Page (Default on Payment Prediction)



We welcome your feedback and inquiries! 💡 💬 We're here to help! 🤝

Don't hesitate to get in touch if you need clarification or have any thoughts to share.😊

About Us

This is an AI-powered application desinged to help identify whether a particular credit card holder default or not and the transection is fraudulent or not, to help the lenderers. Our main goal is to leverage the power of machine learning and data analysis to do it.

Our team consists of data scientists and software engineers who are passionate about using technology to make positive impact in the industry.

Contact us

For more information or inquiries, please contact us at:

📧 tapankheni10304@gmail.com

📞 +91 6354541592

Connect with us

Figure 3.3.5 About Us Page

4. LEARNING OUTCOME DAYWISE

Date	Work to carry out	Learning Outcome
24/06//2024	Installed Required Libraries	Installation of Python libraries (Pandas, Scikit-learn)
29/06/2024	Dataset Collection and Preprocessing	Understanding data cleaning and preprocessing steps
01/07/2024	Exploratory Data Analysis (EDA)	Insights into Data Patterns and Features Importance
02/07/2024	Trained Initial Machine Learning Models for Fraud Detection	Learned about Models like SVM, Random Forest etc.
03/07/2024	Trained Models for Customer Default Prediction	Comparative Analysis of Model Performance
04/07/2024	Hyperparameter Tuning for Best Models	Understanding of Model Optimization and Tuning Techniques
05/07/2024	Developed Streamlit Web Application	Created Basic Web Interface for User Interaction
08/07/2024	Created Cognitive Training Page ii Web Application	Enabled Users to Train Models Dynamically
09/07/2024	Created Prediction Page for Fraud Detection and Default Prediction	Implemented Real-time Prediction Based on User Input
10/07/2024	Finalized UI / UX for Web Application	Improved User Interface and Experience
11/07/2024	Evaluation and Testing	Performed Comprehensive Testing and Achieved 99% Accuracy
12/07/2024	-	-

5. CONCLUSION & FUTURE WORK

5.1 OVERALL ANALYSIS OF PROJECT VIABILITIES

1. **Technical Viability:**

   The system is built using robust, well-established machine learning algorithms and technologies. With high model accuracy (around 99% for both fraud detection and default prediction), the project shows great potential for real-world deployment. The use of Streamlit for web development ensures that the application is scalable, user-friendly, and accessible even for non-technical users.

2. **Business Viability**:

   From a business standpoint, this project addresses critical challenges for financial institutions, such as detecting fraudulent transactions and identifying high-risk customers who are likely to default on payments. Given the high accuracy achieved, this project can significantly reduce fraud losses and improve customer risk assessment, making it an attractive solution for banks, credit card companies, and financial institutions. Additionally, the dual-purpose nature of the system (fraud and default) provides versatility, potentially saving costs by consolidating risk management efforts.

5.2 PROBLEMS ENCOUNTERED AND POSSIBLE SOLUTION

1. **Data Imbalance:**

   One of the main challenges was handling the imbalance in the datasets, particularly for default detection, as the number of customers who default on their payments are fewer than the others.

   Solution:

   - Techniques such as oversampling the minority class (e.g., SMOTE) or undersampling the majority class were used to address the imbalance issue. Additionally, metrics like Precision-Recall and F1-Score were used for evaluating models rather than just accuracy.

2. **Model Overfitting**:
   During model training, especially with complex algorithms like Random Forest and XGBoost, there was a risk of overfitting the models to the training data, resulting in poor generalization on unseen data.
   Solution:
   - Cross-validation and regularization techniques (such as L1 and L2 regularization) were implemented to prevent overfitting. Hyperparameter tuning was also performed to optimize the models for better generalization.

3. **Integration of Web Application**:
   Integrating the machine learning models with the web interface was challenging, particularly in terms of ensuring efficient data processing, model loading, and real-time predictions.
   Solution:
   - The problem was resolved by optimizing data flow and preprocessing within the web application and using efficient model serialization methods such as Pickle or Joblib for loading the trained models.

5.3 SUMMARY OF PROJECT WORK

The Financial Firewall: Credit Card Security AI project aimed to develop an intelligent system that tackles two crucial aspects of financial security: detecting fraudulent transactions and predicting customer defaults. The project involved:

- **Data Collection and Preprocessing:** Curated two datasets—one for fraud detection and one for customer default prediction—and performed thorough preprocessing, including handling missing values and addressing class imbalance.
- **Model Training and Evaluation:** Multiple machine learning models were trained and evaluated for both tasks, with Random Forest and XGBoost being the most successful in terms of accuracy and performance.

- **Web Application Development:** The project culminated in a web-based application built using Streamlit, enabling users to interact with the models for training and predictions. The app featured pages for model training, prediction, and general information, making it user-friendly for non-technical users.

5.4 LIMITATIONS AND FUTURE ENHANCEMENT

Limitations:

1. **Data Dependence:**

   The accuracy and effectiveness of the models heavily depend on the quality and size of the training datasets. In real-world applications, these datasets may vary, and the models could require constant updating to maintain accuracy.

2. **Real-Time Scalability**:

   The current deployment may struggle with handling large-scale real-time data streams due to the nature of the machine learning models and the deployment platform's limitations. A more sophisticated backend, such as AWS Lambda or Apache Kafka, may be necessary to process real-time transactions in bulk.

Future Enhancement:

1. **Incorporation of Advanced Algorithms**:

   Future iterations could integrate deep learning techniques such as Recurrent Neural Networks (RNN) or Graph Neural Networks (GNN) to enhance the prediction capabilities, especially for fraud detection.

2. **Real-Time Processing**:

   To make the system more viable for financial institutions, it could be expanded to support real-time transaction processing. Implementing streaming technologies and low-latency models would enable faster fraud detection and prevention.

3. **Extended Functionality**:

   Adding more features such as customer segmentation, loan eligibility prediction, and personalized financial advisory could broaden the scope of the project and provide additional value to financial institutions.

4. I**mproved User Interface and Experience**:

   The web application's UI/UX can be enhanced to make it more visually appealing and intuitive, especially for non-technical users. Adding data visualization tools for fraud trends and default predictions over time would provide better insights.

5. **Integration with APIs**:

   The system could be integrated with existing banking systems via APIs, enabling seamless integration and real-time fraud prevention in financial institutions.

REFERENCES

1. https://scikit-learn.org/stable/ (For Learning Scikit-learn)
2. https://streamlit.io/ (For Learning Streamlit)
3. https://www.python.org/ (For Python)
4. https://catboost.ai/ (To Implement CatBoost Model)
5. https://xgboost.readthedocs.io/en/latest/ (To Implement XGBoost)
6. https://www.kaggle.com/ (To get the Dataset)