Project Report

Online Fraud Detection System for the Indian Landscape

By detecting fraudulent transactions, this model helps protect users and businesses in the Indian digital landscape, fostering a safer and more trustworthy online environment

Name – Roshan Rateria

1. Introduction

1.1 Project Overview

In the rapidly evolving digital landscape of India, online transactions have witnessed exponential growth. While this surge brings immense convenience and opportunities, it simultaneously exposes users and businesses to a heightened risk of fraudulent activities. To combat this escalating threat, the development of robust and adaptive fraud detection systems has become imperative.

This project focuses on building a comprehensive online fraud detection system tailored specifically for the Indian context. Leveraging machine learning techniques and a unique dataset reflecting the diverse nature of online fraud in India, the system aims to accurately identify and classify different types of fraudulent transactions.

This report details the project's journey, encompassing the problem definition, solution ideation, design and development process, performance evaluation, and future scope. The system's potential impact on enhancing the security of online transactions within the Indian digital ecosystem will also be discussed.

1.2 Purpose

The primary purpose of this project is to create a practical and effective solution for combating online fraud in India. The system aims to:

- **Accurately Detect Fraud:** Identify fraudulent online transactions with a high degree of accuracy, minimizing false positives and negatives.
- **Classify Fraud Types:** Categorize fraudulent transactions into specific types (e.g., phishing, scams, identity theft, payment card fraud) to provide actionable insights.
- **Enhance Security:** Strengthen the security of online platforms and transactions, protecting users and businesses from financial losses and reputational damage.
- **Improve User Trust:** Foster greater trust and confidence in the Indian digital economy by providing a safer environment for online transactions.
- **Support Law Enforcement:** The system's insights can potentially assist law enforcement agencies in investigating and mitigating online fraud activities.

1.3 Project Scope

This project encompasses the following key areas:

- **Data Analysis:** In-depth analysis of a comprehensive Indian online fraud dataset to understand fraud patterns and trends.
- **Feature Engineering:** Selection and transformation of relevant features from the dataset to train and optimize machine learning models.
- **Model Development:** Building and evaluating multiple machine learning models for fraud detection and fraud type classification.
- **Model Optimization:** Fine-tuning model parameters and experimenting with different algorithms to maximize performance.
- **User Interface Development:** Creating a user-friendly interface using Gradio to demonstrate the system's functionality and provide predictions.
- **Documentation:** Providing detailed documentation of the project, including the code, architecture, and performance metrics.

1.4 Project Deliverables

The key deliverables of this project include:

- **Functional Fraud Detection System:** A working prototype of the online fraud detection system implemented in Python.
- **Interactive Gradio App:** A user-friendly Gradio application that allows users to input transaction details and receive fraud predictions.
- **Comprehensive Project Report:** This document, providing a detailed overview of the project, its methodology, findings, and future directions.
- **Source Code Repository:** A GitHub repository hosting the project's source code, allowing for transparency, collaboration, and further development.

1.5 Project Methodology

This project follows a structured methodology incorporating elements of data science, machine learning, and software engineering:

1. **Problem Definition:** Clearly defining the problem of online fraud in India and the specific challenges addressed by this project.
2. **Data Acquisition and Exploration:** Obtaining the relevant dataset and performing exploratory data analysis to understand its characteristics, patterns, and relationships within the data.
3. **Data Preprocessing and Feature Engineering:** Preparing the data for modeling, including handling missing values, converting categorical variables, and engineering new features.
4. **Model Selection and Training:** Choosing appropriate machine learning algorithms based on the problem definition and dataset, followed by training the models using the preprocessed data.
5. **Model Evaluation and Optimization:** Evaluating the performance of trained models using relevant metrics, and fine-tuning hyperparameters to improve accuracy and generalization.
6. **User Interface Development:** Designing and developing a user-friendly interface using Gradio to demonstrate the model's functionality.
7. **Documentation and Reporting:** Documenting the entire project lifecycle, from problem definition to solution implementation and evaluation.

2. Literature Survey

2.1 Existing Problem

The surge in online transactions in India has been accompanied by a parallel rise in online fraud. Fraudsters are constantly evolving their tactics, exploiting vulnerabilities in systems and leveraging social engineering to deceive users. The impact of online fraud is multifaceted, resulting in significant financial losses for individuals, businesses, and the overall economy.

2.1.1 Prevalence of Online Fraud in India

According to a report by the Reserve Bank of India (RBI), there has been a significant increase in digital fraud cases in recent years. Factors contributing to this surge include:

- **Increased Smartphone and Internet Penetration:** A larger pool of potential victims with varying levels of digital literacy.
- **Evolving Fraud Methods:** Sophisticated techniques like phishing, malware, and social engineering attacks.
- **Data Breaches:** Compromised user data from various sources empowers fraudsters with valuable information for targeted attacks.
- **Lack of Awareness:** Many users lack awareness about online security practices, making them vulnerable to fraud.

2.1.2 Types of Online Fraud

The Indian landscape witnesses a diverse range of online fraud, including:

- **Payment Card Fraud:** Unauthorized use of credit/debit card information for purchases or cash withdrawals.
- **Phishing:** Deceptive attempts to obtain sensitive information like usernames, passwords, and financial details by posing as trustworthy entities.
- **Malware:** Malicious software designed to gain unauthorized access to devices or networks, often used to steal data or disrupt operations.
- **Identity Theft:** Using another person's identity to open accounts, make transactions, or commit other crimes.
- **Advance Fee Scams:** Tricking individuals into paying upfront fees for goods or services that are never delivered.
- **E-commerce Fraud:** Fraudulent activities related to online shopping, such as fake websites, non-delivery of products, or counterfeit goods.

2.1.3 Challenges in Traditional Fraud Detection Methods

Traditional rule-based fraud detection systems often struggle to keep pace with the evolving tactics of fraudsters. Limitations include:

- **High False Positives:** Rule-based systems often generate many false positives, flagging legitimate transactions as suspicious and causing inconvenience to users.
- **Inability to Adapt:** These systems struggle to adapt to new fraud patterns and emerging threats, requiring frequent manual updates.
- **Limited Scope:** Rule-based systems are often limited in their ability to detect complex fraud schemes that involve multiple variables and intricate patterns.

2.2 References

The following resources provide valuable insights into the landscape of online fraud in India:

- Reserve Bank of India (RBI) Annual Reports and Publications on Digital Payments Security
- Data Security Council of India (DSCI) Reports and Analyses
- National Crime Records Bureau (NCRB) Data on Cybercrime
- Academic Research Papers on Machine Learning for Fraud Detection

The following Dataset was used in this project:

- https://www.kaggle.com/datasets/kumarperiya/comprehensive-indian-online-fraud-dataset

The following sites are helpful for understanding the Project:

- https://scikit-learn.org/1.5/modules/generated/sklearn.linear_model.LogisticRegression.html
- https://scikit-learn.org/stable/modules/tree.html
- https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html
- https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
- https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html
- https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html
- https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc

2.3 Problem Statement Definition

There is a pressing need for a robust and adaptive online fraud detection system in India that can effectively address the following:

- **High Accuracy:** The system should accurately identify fraudulent transactions with minimal false positives and false negatives.
- **Adaptability:** It should be able to learn and adapt to evolving fraud patterns and emerging threats.
- **Comprehensiveness:** The system should be able to detect and classify various types of online fraud prevalent in India.
- **Real-Time Capability:** It should be capable of analyzing transactions in real-time to enable immediate fraud prevention measures.
- **User-Friendliness:** The system should have an intuitive interface that allows for easy interpretation of results and actionable insights.

This project addresses the defined problem by leveraging machine learning techniques and a unique dataset specific to the Indian context. The goal is to develop a solution that not only enhances security but also contributes to building a more trustworthy and resilient digital ecosystem in India.

3. Ideation & Proposed Solution

3.1 Empathy Map Canvas

To ensure the development of a user-centric and effective solution, an empathy map canvas was created to understand the perspectives and needs of key stakeholders impacted by online fraud. This exercise focused on two primary stakeholders:

- **E-commerce Business Owners (like David):** Individuals responsible for online businesses and safeguarding customer transactions.
- **Online Consumers:** Individuals making online purchases and potentially vulnerable to fraudulent activities.

Empathy Map Canvas for E-commerce Business Owners:

- **What They See:** Rising online fraud rates, evolving fraud tactics, customer distrust due to security breaches, competitive pressure to provide a safe online environment.
- **What They Hear:** Customer complaints about fraudulent transactions, news about data breaches and security vulnerabilities, recommendations for fraud prevention strategies.
- **What They Think & Feel:** Concerned about financial losses, reputational damage, customer churn, the complexity of implementing robust security measures.
- **What They Say & Do:** Emphasize security measures, invest in fraud prevention tools, educate customers about safety practices.
- **Pain Points:** Financial losses due to fraud, damage to brand reputation, customer dissatisfaction and loss of trust, the cost and complexity of implementing effective security measures.
- **Gain Points:** Reduced fraud losses, improved customer trust and loyalty, enhanced brand reputation, a more secure and competitive business environment.

Empathy Map Canvas for Online Consumers:

- **What They See:** News about online scams, warnings about phishing attempts, concerns about data privacy and security.

- **What They Hear:** Stories from friends and family about online fraud experiences, advice on safe online shopping practices.
- **What They Think & Feel:** Anxious about becoming a victim of fraud, concerned about the safety of their personal and financial information, frustrated with complicated security procedures.
- **What They Say & Do:** Look for secure payment options, hesitate to share sensitive information online, verify website authenticity.
- **Pain Points:** Fear of financial loss and identity theft, inconvenience of security measures, lack of trust in online platforms, frustration with fraudulent experiences.
- **Gain Points:** Peace of mind when making online transactions, confidence in the security of online platforms, a seamless and trustworthy online shopping experience.

3.2 Ideation & Brainstorming

Based on the insights gathered from the empathy map canvas and the problem definition, a brainstorming session was conducted to generate potential solutions for an effective online fraud detection system. The following ideas were explored:

3.2.1 Transaction Data Analysis

- **Anomaly Detection:** Implement unsupervised machine learning algorithms to identify unusual patterns in transaction data that deviate from expected behavior.
- **Rule-Based Filters:** Develop a set of rules based on expert knowledge and historical fraud patterns to flag suspicious transactions.
- **Time Series Analysis:** Analyze transaction data over time to detect sudden spikes, unusual frequencies, or other temporal anomalies.
- **Velocity Checks:** Monitor the speed and frequency of transactions from a single account or card to identify potentially fraudulent activity.

3.2.2 User Profile and Behavior Analysis

- **Behavioral Biometrics:** Analyze user behavior patterns, such as typing speed, mouse movements, and navigation patterns, to identify anomalies that could indicate fraud.
- **Device Fingerprinting:** Create unique device fingerprints based on hardware and software configurations to identify suspicious devices or multiple accounts originating from the same device.
- **Geolocation Tracking:** Compare transaction locations with user profiles and historical data to identify inconsistencies and potential fraud.

3.2.3 External Data Enrichment

- **Blacklist/Whitelist Integration:** Integrate with fraud prevention databases and blacklists to flag known fraudulent accounts, devices, or IP addresses.
- **Social Media Analysis:** Analyze social media data to identify potential fraud rings, scams, or phishing attempts.
- **Credit Bureau Data Integration:** Access credit bureau data to assess user risk profiles and identify potentially fraudulent activities.

3.2.4 Machine Learning Models

- **Supervised Learning:** Train supervised machine learning models like Logistic Regression, Support Vector Machines (SVM), Decision Trees, Random Forests, and Neural Networks on labeled datasets of fraudulent and non-fraudulent transactions.
- **Unsupervised Learning:** Use unsupervised learning techniques like clustering and anomaly detection to identify suspicious patterns in data without prior labeling.
- **Ensemble Methods:** Combine multiple machine learning models to leverage their individual strengths and improve overall prediction accuracy.

3.3 Proposed Solution

After careful consideration of the brainstormed ideas and the project's specific requirements, the proposed solution involves a multi-faceted approach:

1. **Data-Driven Fraud Detection:** Utilize the "Inclusive Indian Fraud Dataset" to train and evaluate multiple machine learning models for both fraud detection (binary classification) and fraud type classification (multi-class classification).
2. **Focus on Indian Context:** The solution will specifically address the types of fraud prevalent in India, considering factors like common payment methods, e-commerce platforms, and fraud tactics observed in the region.
3. **User-Friendly Interface:** Develop a user-friendly Gradio application that allows users to input transaction details and receive real-time fraud predictions. This interface will also provide insights into the factors influencing the prediction, enhancing transparency and user understanding.

By combining a data-driven approach with a focus on the Indian context and a user-friendly interface, this project aims to create a practical and impactful solution for combating online fraud in India.

4. Requirement Analysis

After brainstorming potential solutions and solidifying the project's direction, it's crucial to define the specific requirements that will guide the system's development. These requirements will ensure the system effectively addresses the problem of online fraud in the Indian context while meeting the needs of its intended users.

4.1 Functional Requirements

Functional requirements outline the system's core capabilities and how it should behave in response to different inputs and scenarios. These requirements form the foundation of the system's functionality.

4.1.1 Data Ingestion and Preprocessing:

- **F.1.1.1:** The system shall be able to ingest data from the "Inclusive Indian Fraud Dataset" provided in CSV format.
- **F.1.1.2:** The system shall handle missing data appropriately, either through imputation or removal, depending on the feature and its importance.
- **F.1.1.3:** The system shall preprocess categorical features using suitable encoding techniques (e.g., one-hot encoding, label encoding) to prepare them for machine learning algorithms.
- **F.1.1.4:** The system shall provide options for feature scaling and normalization to handle variations in feature ranges and improve model performance.

4.1.2 Model Training and Evaluation:

- **F.1.2.1:** The system shall allow users to select from a range of suitable machine learning algorithms for fraud detection, including but not limited to:
    - Logistic Regression
    - Decision Tree
    - Random Forest
    - Support Vector Machine (SVM)
- **F.1.2.2:** The system shall enable users to train the selected model(s) on the preprocessed dataset, specifying the target variable (fraudulent or not fraudulent) and any desired hyperparameters.
- **F.1.2.3:** The system shall evaluate the trained model's performance using appropriate metrics for binary classification tasks, including:

  - Accuracy
  - Precision
  - Recall
  - F1-Score
  - AUC-ROC (Area Under the Receiver Operating Characteristic Curve)
- **F.1.2.4:** The system shall provide visualizations of the model's performance, such as confusion matrices and ROC curves, to aid in understanding and interpreting the results.

4.1.3 Fraud Prediction:

- **F.1.3.1:** The system shall provide an interface (using Gradio) where users can input transaction details, including:
  - Transaction Amount
  - Card Type
  - Location
  - Purchase Category
  - Customer Age
  - Time of Day
- **F.1.3.2:** The system shall use the trained model to generate a fraud prediction (fraudulent or not fraudulent) based on the input transaction details.
- **F.1.3.3:** The system shall display the prediction result to the user along with a confidence score indicating the model's certainty in its prediction.

4.1.4 Fraud Type Classification:

- **F.1.4.1:** The system shall allow users to select from a range of suitable multi-class classification algorithms to predict the type of fraud, including:
  - Random Forest
  - Gradient Boosting
- **F.1.4.2:** The system shall train the selected model on the preprocessed dataset using the "fraud_type" column as the target variable.
- **F.1.4.3:** The system shall predict the most likely fraud type for a given transaction based on the user's input.
- **F.1.4.4:** The system shall display the predicted fraud type to the user, along with the probability scores for each potential fraud type.

4.2 Non-Functional Requirements

Non-functional requirements define the system's quality attributes and constraints, ensuring it meets user expectations and operational standards.

4.2.1 Performance:

- **NF.1.1.1:** The system should provide prediction results within a reasonable timeframe (ideally less than 2 seconds) to ensure a smooth user experience.
- **NF.1.1.2:** The system should be able to handle a moderate volume of concurrent user requests without significant performance degradation.

4.2.2 Usability:

- **NF.1.2.1:** The Gradio interface should be intuitive and user-friendly, allowing users with minimal technical knowledge to easily input data and interpret results.
- **NF.1.2.2:** The system should provide clear instructions and explanations for each input field and output result.

4.2.3 Security:

- **NF.1.3.1:** The system should handle user data responsibly and comply with relevant data privacy regulations.

- **NF.1.3.2:** Sensitive data like transaction details should be transmitted securely (using HTTPS) to prevent unauthorized access.

4.2.4 Maintainability:

- **NF.1.4.1:** The system's codebase should be well-structured, documented, and modular to facilitate future maintenance and updates.
- **NF.1.4.2:** The system should be designed to accommodate the integration of new data sources and the addition of new machine learning models as needed.

These functional and non-functional requirements provide a clear roadmap for the design, development, and implementation of the online fraud detection system. By adhering to these requirements, the project aims to create a solution that is not only effective but also user-friendly, scalable, and aligned with the specific needs of the Indian digital landscape.

5. Project Design

This section outlines the design of the online fraud detection system, encompassing the system's architecture, data flow, and user interface. These design elements ensure the system effectively meets the requirements defined in the previous section while providing a user-friendly and robust solution.

5.1 Data Flow Diagrams & User Stories

5.1.1 Data Flow Diagram (DFD)

1. **Data Input:** The system receives raw transaction data from the "Inclusive Indian Fraud Dataset" (CSV format).
2. **Data Preprocessing:** The data undergoes cleaning, transformation, and feature engineering steps. This includes handling missing values, encoding categorical features, and scaling numerical features.
3. **Model Training:** The preprocessed data is used to train the selected machine learning models (fraud detection and fraud type classification).
4. **User Input:** Users interact with the Gradio interface, providing transaction details for prediction.
5. **Fraud Prediction:** The system uses the trained models to generate fraud predictions and confidence scores based on user input.
6. **Fraud Type Prediction:** If a transaction is predicted as fraudulent, the system predicts the most likely fraud type.
7. **Output Display:** The Gradio interface displays the prediction results, confidence scores, and any additional relevant information to the user.

5.1.2 User Stories

User stories help to capture the system's functionality from the user's perspective:

- **User Story 1 (E-commerce Business Owner):**
    - As an e-commerce business owner, I want to be able to upload my transaction data to the system to train a fraud detection model specific to my business.
- **User Story 2 (Risk Analyst):**
    - As a risk analyst, I want to be able to select from different machine learning models, adjust model parameters, and evaluate the model's performance to optimize fraud detection.
- **User Story 3 (Customer Support Agent):**
    - As a customer support agent, I want to use the system to quickly check if a transaction is potentially fraudulent so I can take appropriate action.
- **User Story 4 (Online Consumer):**

- As an online consumer, I want to understand the factors that contributed to a fraud prediction so I can be more aware of potential risks.

5.2 Solution Architecture

The online fraud detection system follows a modular architecture, separating concerns and allowing for flexibility and scalability:

5.2.1 Data Layer:

- **Dataset:** The "Inclusive Indian Fraud Dataset" serves as the primary data source.
- **Data Preprocessing Module:** Handles data cleaning, transformation, and feature engineering.

5.2.2 Model Layer:

- **Model Training Module:** Allows for training and managing multiple machine learning models.
- **Model Storage:** Stores the trained models for future use.

5.2.3 Prediction Engine**:**

- **Fraud Detection Model:** Predicts the likelihood of a transaction being fraudulent.
- **Fraud Type Classification Model:** Classifies the type of fraud for predicted fraudulent transactions.

5.2.4 User Interface Layer:

- **Gradio Application:** Provides an interactive interface for user input, model selection, prediction results, and visualizations.

5.2.5 Technology Stack:

- **Programming Language:** Python
- **Machine Learning Libraries:** Scikit-learn, TensorFlow/PyTorch (optional for neural networks)
- **User Interface Framework:** Gradio
- **Data Visualization:** Matplotlib, Seaborn
- **Cloud Platform:** (Optional) AWS, Google Cloud Platform, Azure

5.3 User Interface Design

The Gradio application will provide a clean and intuitive interface with the following features:

- **Data Input Form:** Clear and labeled fields for users to enter transaction details.
- **Model Selection Dropdown:** Allow users to choose between the trained fraud detection models.
- **Prediction Results Display:** Present the prediction outcome (fraudulent/not fraudulent) along with a confidence score.
- **Fraud Type Breakdown (Optional):** If a transaction is flagged as fraudulent, display a breakdown of the predicted fraud type probabilities.
- **Feature Importance Visualization (Optional):** Highlight the key features contributing to the prediction, helping users understand the model's decision-making process.

By following this design, the system will be equipped to effectively address the challenges of online fraud detection in the Indian context. The combination of machine learning, a user-friendly interface, and a focus on the specific types of fraud prevalent in India will contribute to a more secure and trustworthy online environment.

6. Project Planning & Scheduling

To ensure the timely and successful completion of the online fraud detection system, this section outlines the project's technical architecture, sprint planning, estimation, and a proposed delivery schedule.

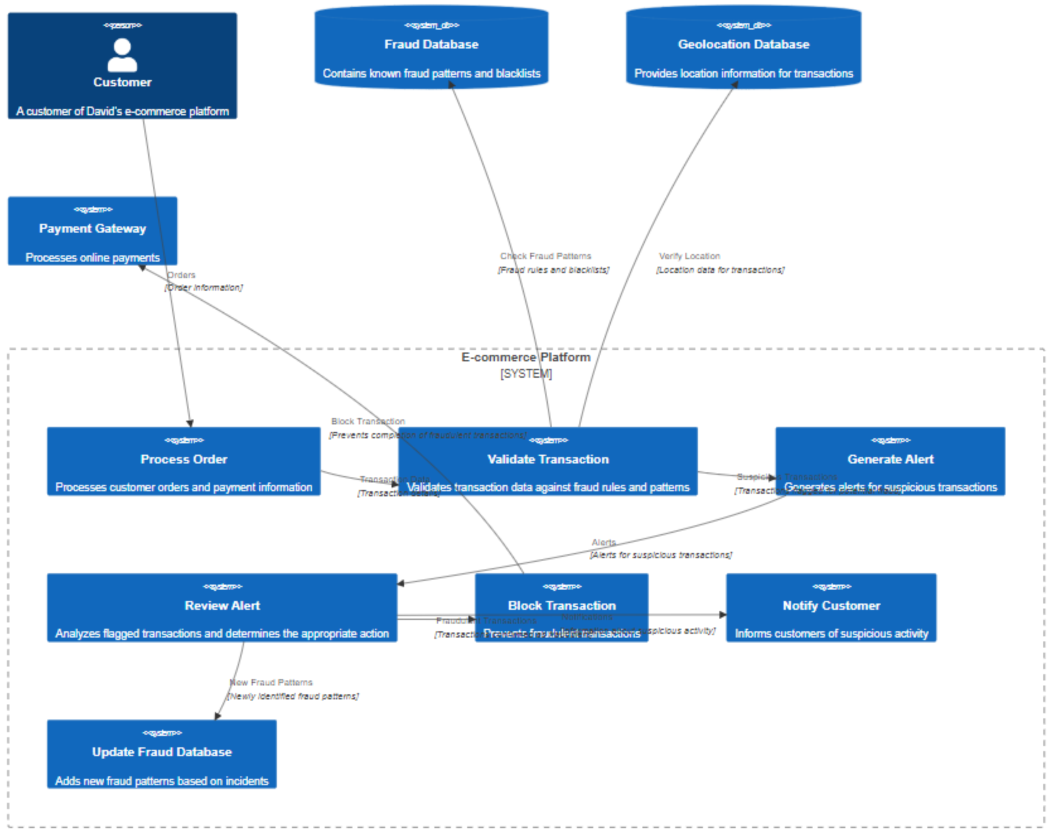
6.1 Technical Architecture

The system will be built using a modular architecture, employing open-source technologies for flexibility and cost-effectiveness. The architecture leverages a cloud-based approach for scalability and ease of deployment.

6.1.1 System Components:

1. **Data Storage:**
    - **AWS S3 Bucket:** Used to store the raw "Inclusive Indian Fraud Dataset" and any additional data sources. S3 provides secure, scalable, and cost-effective storage.
2. **Data Processing and Model Training:**
    - **AWS EC2 Instance:** A virtual server instance will be provisioned to handle data preprocessing, feature engineering, model training, and evaluation.
    - **Python Environment:** The EC2 instance will have a configured Python environment with necessary libraries like Pandas, Scikit-learn, and potentially TensorFlow/PyTorch for more complex models.
3. **Model Deployment and Prediction:**
    - **Hugging Face :** A serverless compute service used to deploy the trained machine learning models. Lambda functions automatically scale based on demand, providing a cost-effective way to handle predictions.
4. **User Interface:**
    - **Gradio App:** Developed using Python and Gradio, the app will be hosted on a separate Hugging face Spaces, providing a public URL for user access.

6.1.2 Architecture Diagram:



<<person>>
Customer
A customer of David's e-commerce platform
<<system_db>>
Fraud Database
Contains known fraud patterns and blacklists
<<system_db>>
Geolocation Database
Provides location information for transactions
<<system>>
Payment Gateway
Processes online payments
Orders
[Order information]
Check Fraud Patterns
[Fraud rules and blacklists]
Verify Location
[Location data for transactions]
E-commerce Platform
[SYSTEM]
Block Transaction
[Prevents completion of fraudulent transactions]
<<system>>
Process Order
Processes customer orders and payment information
Transaction Data
[Transaction details]
<<system>>
Validate Transaction
Validates transaction data against fraud rules and patterns
Suspicious Transactions
<<system>>
Generate Alert
Generates alerts for suspicious transactions
Alerts
[Alerts for suspicious transactions]
<<system>>
Review Alert
Analyzes flagged transactions and determines the appropriate action
Fraudulent Transactions
<<system>>
Block Transaction
Prevents fraudulent transactions
Notifications
[Information about suspicious activity]
<<system>>
Notify Customer
Informs customers of suspicious activity
New Fraud Patterns
[Newly identified fraud patterns]
<<system>>
Update Fraud Database
Adds new fraud patterns based on incidents

6.2 Sprint Planning & Estimation

The project will follow an Agile methodology with a series of 2-week sprints to ensure flexibility and iterative development.

Sprint 1 (Data Preparation and Exploration):

- **Tasks:**
  - Set up the development environment (EC2 instance, Python libraries).
  - Download and store the "Inclusive Indian Fraud Dataset" in S3.
  - Perform data exploration and visualization to understand the dataset's characteristics.
- **Estimated Effort:** 10 working days

Sprint 2 (Feature Engineering and Model Training):

- **Tasks:**
  - Implement data preprocessing and feature engineering pipelines.
  - Train the fraud detection model using the selected algorithm(s).
  - Evaluate the model's performance and fine-tune as needed.
- **Estimated Effort:** 10 working days

Sprint 3 (Fraud Type Classification and API Development):

- **Tasks:**
  - Train the fraud type classification model.
  - Develop the API endpoints for fraud prediction and fraud type prediction using AWS Lambda and API Gateway.
- **Estimated Effort:** 10 working days

Sprint 4 (Gradio App Development and Integration):

- **Tasks:**
  - Design and develop the user interface using Gradio.

    - Integrate the Gradio app with the API endpoints to enable real-time predictions.
    - Perform user acceptance testing (UAT) to ensure usability and functionality.
- **Estimated Effort:** 10 working days

6.3 Sprint Delivery Schedule

Sprint	Start Date	End Date	Deliverables
Sprint 1	23 Sept 2024	24 Sept 2024	Data exploration report, initial code repository setup
Sprint 2	24 Sept 2024	25 Sept 2024	Trained fraud detection model, model evaluation report
Sprint 3	25 Sept 2024	26 Sept 2024	Trained fraud type classification model, functional API endpoints
Sprint 4	26 Sept 2024	27 Sept 2024	Functional Gradio application, integrated with the prediction API, UAT report

This proposed schedule provides a roadmap for the project's development. The Agile methodology allows for flexibility in adapting to unexpected challenges and incorporating user feedback throughout the development process. Regular sprint reviews and communication will ensure the project stays on track and delivers a robust and effective fraud detection solution within the given timeframe.

7. Coding & Solutioning

This section delves into the technical details of the online fraud detection system, showcasing the code implementation and highlighting key features. The models along with label encoders are saved and for predicting future values. While this report provides a concise overview, the complete source code is available in the accompanying GitHub repository.
(Code snippets are omitted in this section for brevity. Refer to the accompanying GitHub repository for the complete source code.)

7.0 Dataset:

7.0.1 Description:

This project utilizes the "Inclusive Indian Fraud Dataset" as its primary data source. This dataset provides a comprehensive view of online fraud activities specifically within the Indian context, making it a valuable resource for building and evaluating fraud detection models tailored to this region.

Key Features:

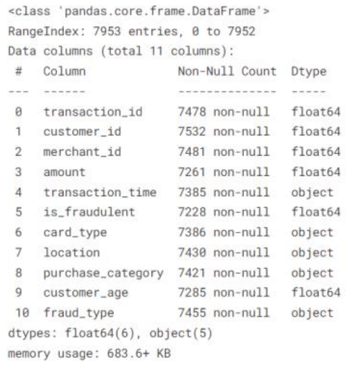
- **Transaction ID:** Unique identifier for each transaction.
- **Customer ID:** Unique identifier for each customer involved.
- **Merchant ID:** Unique identifier for each merchant involved.
- **Transaction Amount:** The monetary value of the transaction.
- **Card Type:** The type of card used for the transaction (e.g., Visa, MasterCard, RuPay).
- **Location:** The geographical location where the transaction occurred (focused on major Indian cities).

- **Purchase Category:** The category of goods or services purchased in the transaction (e.g., Digital, POS).
- **Customer Age:** The age of the customer making the transaction.
- **Transaction Time:** The date and time when the transaction took place.
- **Fraud Type:** Categorical labels indicating the specific type of fraud, including:
    - Malware
    - Payment Card Fraud
    - Identity Theft
    - Scam
    - Phishing
- **Is Fraudulent:** A binary label indicating whether the transaction was fraudulent (1) or not (0).

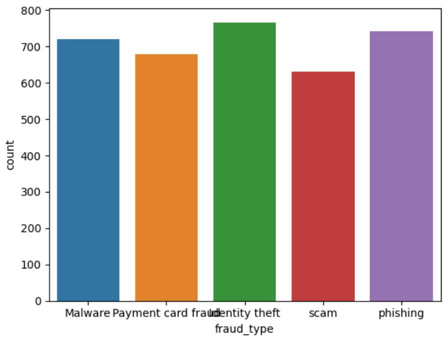
7.0.2 Data Analaysis

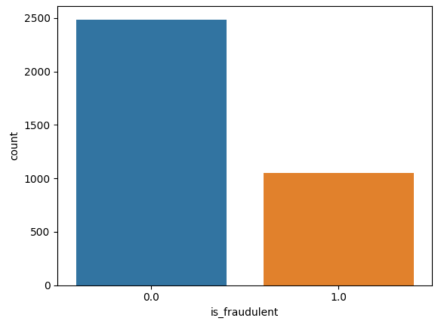
The initial phase of the project involved conducting exploratory data analysis (EDA) to gain insights into the dataset's characteristics, distributions, and potential relationships between variables.

- **Data Summary:** Basic statistics.
- **Data Visualization:** count plots to visualize data distributions.
- **Unique values from categorical Columns:**
    - Card Types : 'MasterCard' ,'Visa' ,'Rupay'
    - Location: ‘Surat', 'Hyderabad' ,'Kolkata', 'Mumbai' ,'Delhi', 'Chennai', 'Jaipur', 'Ahmedabad' ,'Bangalore', 'Pune
    - Purchase Category: 'POS' ,'Digital'
- **Correlation Analysis:** Examine potential correlations between features to understand their relationships and inform feature selection



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7953 entries, 0 to 7952
Data columns (total 11 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   transaction_id     7478 non-null   float64
 1   customer_id        7532 non-null   float64
 2   merchant_id        7481 non-null   float64
 3   amount             7261 non-null   float64
 4   transaction_time   7385 non-null   object
 5   is_fraudulent      7228 non-null   float64
 6   card_type          7386 non-null   object
 7   location           7430 non-null   object
 8   purchase_category  7421 non-null   object
 9   customer_age       7285 non-null   float64
 10  fraud_type         7455 non-null   object
dtypes: float64(6), object(5)
memory usage: 683.6+ KB
```

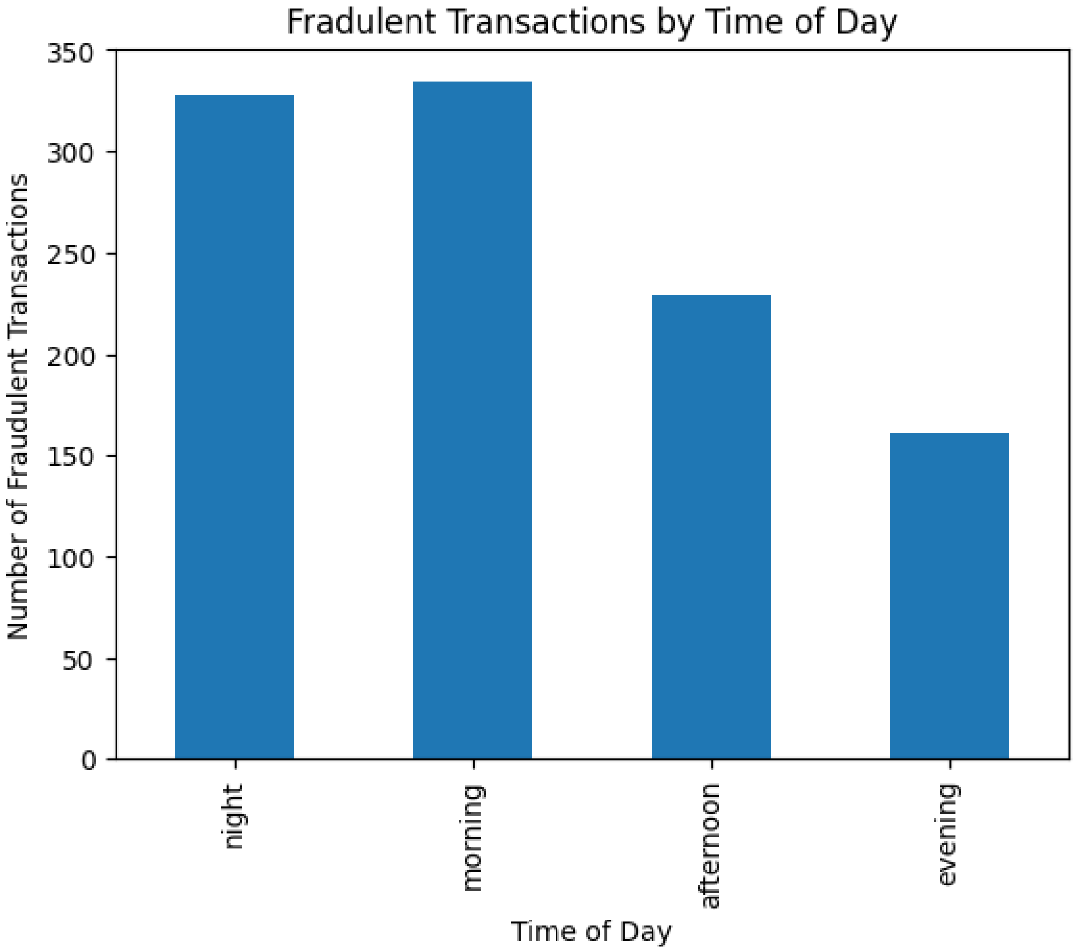




7.0.3 Feature Engineering:

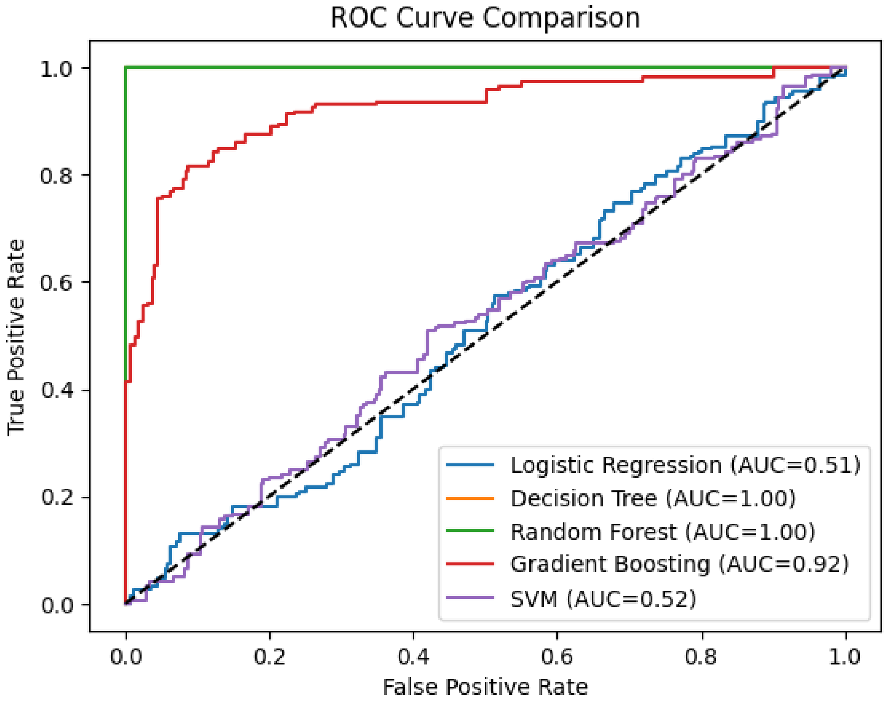
We create a new feature `time_of_day` from the timestamp column to analyse the time of day and if it impacts the times of fraud.It is categorised into night,afternoon,evening,morning.

We Convert all categorical Columns to numeric data using Sklearn LabelEncoders.



7.1 Feature 1: Fraud Detection Model (Binary Classification)

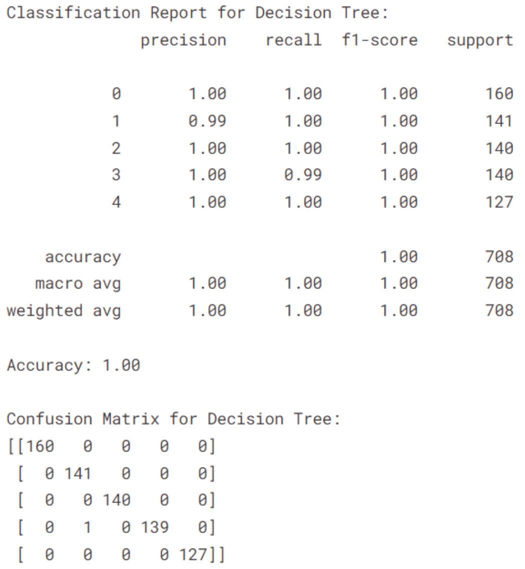
The core functionality of the system lies in its ability to predict whether a given online transaction is fraudulent or not. We tested various models for this binary classification task and found Decision Tree and Random Forest to have 100% accuracy on the Test set.



We choose Decision Tree since it is Simpler than the Random Forest

7.2 Feature 2: Fraud Type Classification (Multi-Class Classification)

Building upon the fraud detection model, this feature aims to categorize predicted fraudulent transactions into specific fraud types present in the dataset. We utilized a Decision Tree Classifier for its ability to handle multiple classes effectively.



```
Classification Report for Decision Tree:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       160
           1       0.99      1.00      1.00       141
           2       1.00      1.00      1.00       140
           3       1.00      0.99      1.00       140
           4       1.00      1.00      1.00       127

    accuracy                           1.00       708
   macro avg       1.00      1.00      1.00       708
weighted avg       1.00      1.00      1.00       708

Accuracy: 1.00

Confusion Matrix for Decision Tree:
[[160   0   0   0   0]
 [  0 141   0   0   0]
 [  0   0 140   0   0]
 [  0   1   0 139   0]
 [  0   0   0   0 127]]
```

7.3 Database Schema (Not Applicable)

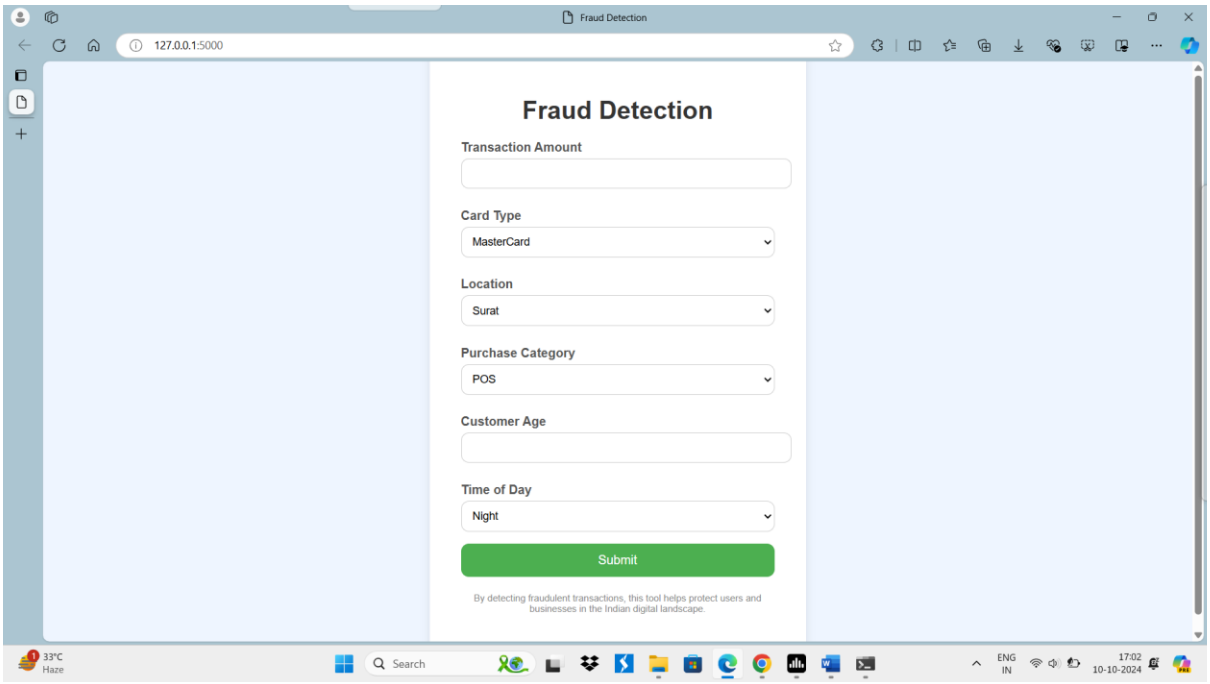
This project does not utilize a separate database management system as the dataset is relatively small and managed efficiently within the Python environment.

7.4 Gradio Application Development

The Gradio application serves as the user interface, enabling users to interact with the trained models and obtain fraud predictions.

- **Input Components:** The interface includes user-friendly input components like numeric fields, dropdown menus, and sliders for users to enter transaction details.
- **Prediction Function:** A dedicated function handles user input, preprocesses it using the defined pipelines, and passes it to the appropriate prediction model.
- **Output Display:** Prediction results, including fraud probability and predicted fraud type (if applicable), are displayed using Gradio's Markdown and Plot components for clear visualization.

Trial Link : https://huggingface.co/spaces/RoAr777/Fraud



Fraud Detection

127.0.0.1:5000

Fraud Detection

Transaction Amount

Card Type: MasterCard

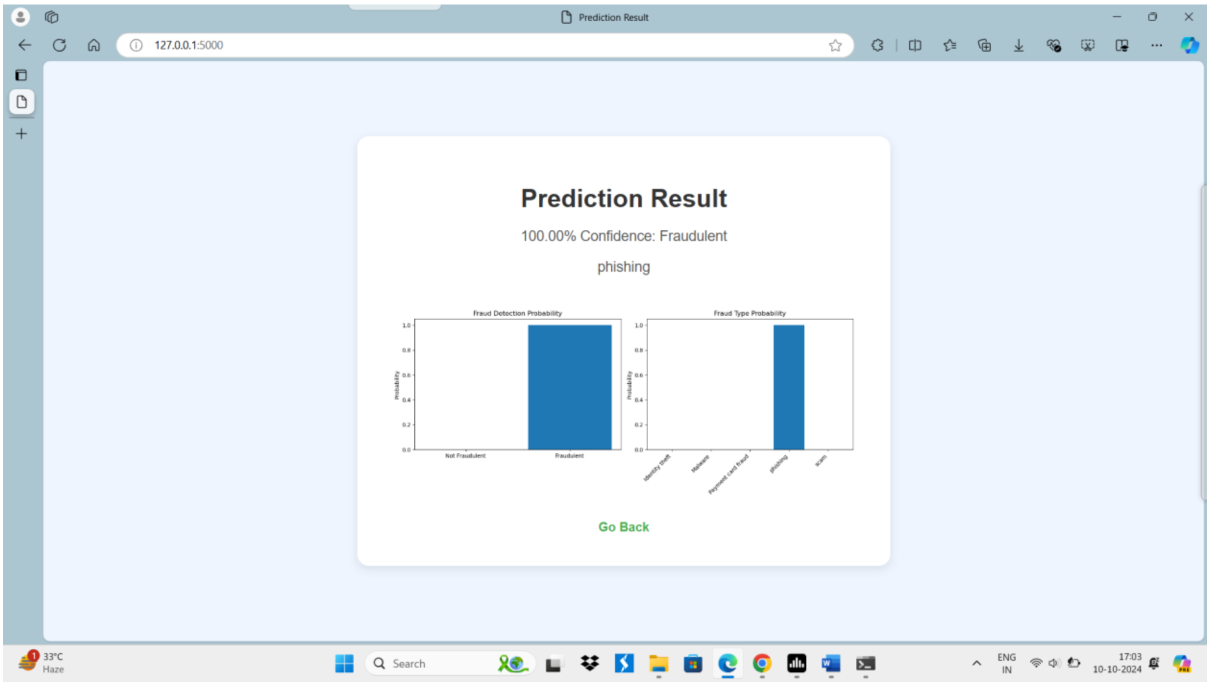
Location: Surat

Purchase Category: POS

Customer Age

Time of Day: Night

By detecting fraudulent transactions, this tool helps protect users and businesses in the Indian digital landscape.



Prediction Result

100.00% Confidence: Fraudulent

phishing

8. Performance Testing

Evaluating the performance of the developed models is crucial to assess their effectiveness in detecting and classifying fraudulent transactions. We employed various metrics and techniques to rigorously test both the fraud detection and fraud type classification models.

8.1 Performance Metrics

- **Accuracy:** Measures the overall correctness of the model's predictions=100%
- **Precision:** Indicates the proportion of correctly identified positive cases (fraudulent transactions) out of all instances predicted as positive=1.00
- **Recall:** Measures the model's ability to correctly identify all positive cases (fraudulent transactions)=1.00
- **F1-Score:** The harmonic mean of precision and recall, providing a balanced measure of the model's performance=1.00

Github Link https://github.com/roshanrateria/Fraud