High-Level Document (HLD)

Fraud Transaction Detection

Revision Number: 1.0

Deepraj Arya

Contents

DOCUMENT VERSION CONTROL

Change Record:

VERSION	DATE	AUTHOR	COMMENTS
0.1	14 JAN 2024	Deepraj Arya	End to End Fraud Transaction Detection

Reviews:

VERSION	DATE	REVIEWER	COMMENTS
0.1	14 JAN 2024		

Approval Status:

VERSION	REVIEW DATE	REVIEWED BY	APPROVED BY	COMMENTS

Abstract

The Fraud Transaction Detection project delves into the evolving landscape of credit card transaction security, aiming to fortify defences against fraudulent activities and mitigate financial losses. This initiative acknowledges the historical challenges faced by similar projects, where traditional approaches often fell short in adapting to the dynamic tactics employed by fraudsters.

Historically, fraud detection in credit card transactions has grappled with the intricacies of identifying patterns in vast datasets, leading to a demand for more sophisticated and adaptable methodologies. This project pioneers an end-to-end solution, employing a comprehensive model pipeline that seamlessly integrates key stages of data processing, transformation, and model training.

To address the shortcomings of previous approaches, the project explores the application of three distinct models—logistic regression, decision trees, and random forests. Drawing from historical data and evolving techniques, these models are carefully tailored to enhance detection accuracy. The system's resilience is further demonstrated by the incorporation of a unified pipeline, streamlining the entire process and ensuring consistency.

The project not only seeks to improve model accuracy but also addresses broader challenges within the domain. Continuous evolution in fraud tactics necessitates a proactive stance, prompting ongoing improvements and adaptations in model architecture. Additionally, the project aims to contribute insights into the ethical considerations surrounding fraud detection, including privacy concerns and responsible use of predictive analytics.

By fostering an understanding of the historical context, current challenges, and ethical dimensions, this project envisions a robust and adaptable framework for fraud transaction detection. The pursuit of continuous improvement, ethical awareness, and technological innovation positions this endeavor at the forefront of advancing credit card transaction security.

1. Introduction

1.1 Introduction

Fraud Transaction Detection project aims to develop a robust system for detecting fraudulent transactions in credit card transactions to enhance security and minimize financial losses. An end-to-end fraud transaction detection system is created for this reason, and the model pipeline idea is used to this task. A robust pipeline is utilised to train the model, which automatically covers all of these stages, after a series of laborious procedures that involve data ingestion, data transformation, and model training. Three different models—logistic regression, decision trees, and random forests—are employed to train the model. Finally, best_model.pkl, the best-performing model, is selected and put in the artefacts folder for future use in prediction. Following this stage, which was completed in a single step using a pipeline, predictions were obtained using the best_model.pkl file and a prediction pipeline.

2. General Description

2.1 Problem Statement

Fraud detection is a set of activities undertaken to prevent money or property from being obtained through false pretenses. Fraud detection is applied to many industries such as banking or insurance. In banking, fraud may include forging checks or using stolen credit cards. Other forms of fraud may involve exaggerating losses or causing an accident with the sole intent for the pay-out. With an unlimited and rising number of ways someone can commit fraud, detection can be difficult to accomplish. Fraud detection is a critical issue for retailers determined to prevent losses and preserve customer trust. Digitalization is one of the major advancements we have in this time. The global market is at the fingertip of each and every individual through Online purchase. Both for the consumers and sellers, online market tends to give more in terms of profit as well as exposure to a larger community. With the increase in digitalization, there is also increase in the fraudulent activities happening in various domains, mainly in the retail domain. These are detrimental to the ecosystem of online transactions. Machine learning provides an intelligent option in dealing with this challenge.

2.2 Proposed Solution

Moving with solution of the problem statement, the primary objective of the Fraud Transaction Detection project is to develop a robust and accurate machine learning model capable of identifying and flagging potentially fraudulent credit card transactions. The project aims to address the following key objectives:

1. Fraud Identification : Detect and identify transactions that are likely to be fraudulent based on historical patterns and anomalies in credit card transactions

2. Real Time Prediction : Enable real-time prediction for incoming credit card transactions, allowing for timely intervention and prevention of potential fraudulent activities.

3. Model Training and Validation : Build and train a machine learning model using historical transaction data. Validate the model's performance using appropriate evaluation metrics to ensure reliable predictions.

4. Model Evaluation : Assess the model's performance on both training and validation datasets, ensuring high accuracy and minimal false positives and false negatives.

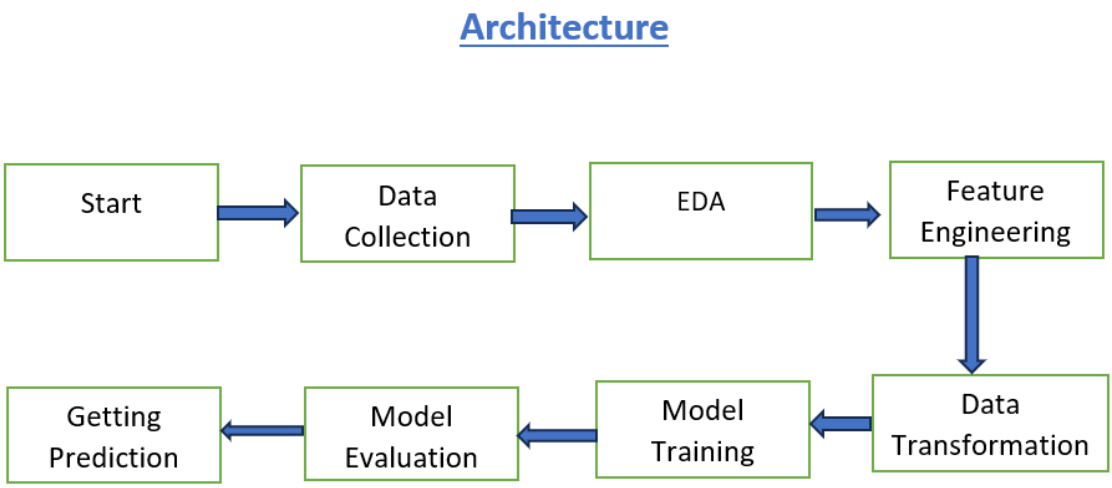
5. User Friendly Interface : Create a user-friendly interface to allow users to input transaction details for real-time prediction and receive immediate feedback on the likelihood of fraud.

2.3 Technical Requirement

- Programming Language : Python is the primary programming language for the project, owing to its extensive libraries and frameworks for machine learning and data processing.
- Version Control : Git is utilized for version control to track changes in the source code and collaborate effectively among team members.
- Data Processing and Analysis : Pandas is employed for efficient data manipulation and analysis, allowing seamless handling of large datasets.
- Machine Learning Libraries :
    - Scikit-learn is essential for implementing machine learning algorithms and model training.
    - Additional libraries such as NumPy, SciPy, and Matplotlib support various aspects of data processing, scientific computing, and visualization.
- Model Training : Logistic Regression, Decision Trees, and Random Forests are implemented using Scikit-learn, providing flexibility and diverse model choices.
- Pipeline Architecture : A robust pipeline architecture is designed, incorporating tools like Scikit-learn's Pipeline module to streamline and automate data processing, transformation, and model training.

3. Design Details

3.1 Process Flow



3.2 Data Preprocessing and Transformation

1. **Data Preparation:** The project begins by loading and preparing the raw credit card transaction data. This includes data cleaning, data type conversion, and handling missing values.

2. Exploratory Data Analysis (EDA): The project conducts exploratory data analysis to gain initial insights into the data to understand the characteristics of the dataset. This includes summary statistics, data distribution, and correlation analysis.

3. Feature Engineering**:** New features like scaled_amound and scaled_time created that may enhance the model's ability to detect fraud and select relevant features based on their importance and contribution to the model.

4. Data Transformation **:** First split the dataset into training and testing sets to reserve a portion of the data for model evaluation. After splitting, choose appropriate machine learning models for fraud detection (e.g., logistic regression, decision trees, random forests).

3.3 Model Training and Evaluation

5. Model Training : Train the selected models using the preprocessed training dataset and find optimize hyperparameters using techniques like grid search or randomized search and then used these hyper parameter to train these models.

6. Model Evaluation **:** Evaluate model performance on the reserved testing dataset. Use metrics such as accuracy to assess the model's effectiveness.

7. Model Validation : Validate the model's performance on new and unseen data. For validation, considered cross-validation techniques to ensure generalization.

3.4 Event Log and Error Handling

By implementing a robust event logging and error handling strategy, the fraud transaction detection system can effectively communicate its status, troubleshoot issues, and maintain operational stability.

- Logging Mechanism
  - Python's logging module or a custom logger, is utilized to categorize log messages into various levels (e.g., INFO, WARNING, ERROR).
- Exception Handling
  - Robust exception handling is integrated into critical sections of the code, encapsulating potential error-prone operations.
  - The try-except block, as demonstrated in the provided code, captures exceptions and logs relevant details using the custom logging mechanism.
- Error Messages
  - Clear and informative error messages are generated to aid in identifying the root cause of issues.
  - Error messages include details such as the type of exception, the specific module or function encountering the error, and any relevant context.

3.5 Performance

The system demonstrated efficient performance throughout the data ingestion, transformation, training, and prediction stages. Timestamped logs provide clear insights into the duration and execution of each stage.

Among all the classification model, we got Decision Tree as the most model with 99.94% accuracy.

3.6 Recommendation for Improvement

- Identify potential bottlenecks or areas for optimization in terms of time or computational resources.
- Consider leveraging parallel processing or distributed computing for large datasets.
- Evaluate the system's scalability and performance under varying data volumes.

4. Dashboards

In the Fraud Transaction Detection project, the implementation of dashboards plays a crucial role in providing users with a user-friendly interface to interact with the system and gain insights into the model's predictions. Flask, a lightweight web application framework, has been utilized to create an API for seamless communication between the backend and the dashboard.

Flask API Implementation : The user interacts with the dashboard through a web browser, accessing the features and insights provided by the Flask API. The intuitive design, coupled with real-time updates, empowers users to make data-driven decisions and proactively address potential fraud scenarios.

5. Conclusion

The Fraud Transaction Detection project has successfully addressed the critical challenge of enhancing security and minimizing financial losses in credit card transactions. Through the development of an end-to-end system employing a robust model pipeline, the project has demonstrated a comprehensive approach to fraud detection.

Followings are key achievements of this project :

- Robust System Architecture:
  The project implemented a well-structured system architecture, comprising data ingestion, data transformation, model training, and prediction stages. The use of a model pipeline streamlined the entire process, ensuring efficiency and consistency.

- Model Training and Selection:
  Three distinct models—Logistic Regression, Decision Trees, and Random Forests—were employed and evaluated. The Decision Tree Classifier emerged as the best-performing model, achieving an impressive accuracy score of 99.94%.

- End-to-End Automation:
  The automation of data transformation and model training through the model pipeline not only reduced manual efforts but also ensured a standardized and reproducible workflow.

- Effective Data Handling:
  The data transformation stage successfully processed and preprocessed the dataset, contributing to the model's robustness and accuracy. The inclusion of timestamped logs provided transparency and traceability throughout the process.

- Training & Prediction Pipeline: The pipelines demonstrated the seamless integration of the trained model into real-world scenarios, enabling the system to make accurate predictions on new data.