Phase-1

Student Name: Vigneshwaran P

Register Number: 422623104032

Institution: Univercity College of Engineering,Panruti

Department:Computer Science and Engineering

Date of Submission:29-04-2025

Problem Statement

The growth of Credit Card (CC) fraud over the last few years necessitates theconstruction of fraud detection models that are both efficient and robust. Thiswork investigated the use of machine learning models, especially ensemblemethods, to improve the detection of CC fraud. We herein introduce anensemble model that combines various classifiers to solve the dataset imbalance problem that is present in most CC datasets. We employed synthetic oversampling and under-sampling techniques in certain machine learning algorithms to tackle the same issue. Online transactions have become an essential aspect of life as universe becomes more technological and every industry leverages theweb to grow enterprises.

Credit cards are a popular form of internet transaction, but with their widespread use comes a significant drawback: credit card fraud. Since banks are unable to screen every transaction, machine learning is essential to identifying credit card fraud. In our research, we used Kaggle to gather a dataset of 2,844,808 credit card transactions from a European Bank Dataset. There are 492 fraudulent transactions in it; to balance the dataset, we proposed hybrid resampling method and for the detection of credit card fraud,Random Forest Algorithm is employed. The evaluation of the model is evaluatedbased on accuracy, precision, recall, and F1-score.

Objectives of the Project

\*The objectives of Credit Card Fraud Detection using machine learning are three and aim finally to secure financial transactions against fraud without causing it.

\*objectives in Credit Card Fraud Detection based on machine learning are to develop accurate, timely, and efficient detection models that are able to accurately pick out fraudulent and genuine transactions with low false positives and false negatives and in accordance with data protection legislation and regulatory needs. Realization of these objectives is highly important for safeguarding financial systems and customers' trust in electronic payment systems.

\***To visualize trends and predictions** in a user-friendly format (e.g., dashboards or charts).

Scope of the Project

* Collection of various credict card frauds from different sources.
* Preprocessing and cleaning of data for model training.
* Implementation of multiple ML algorithms for comparison.
* Visualization and interpretation of results.

Data Sources

* **Source**: UCI Credict Card fraud detection
* **Type**: Public
* **Description**: Contains yearly averaged responses from an array of frauds, along with internet, real-time, and banking.
* **Status**: Static (downloaded once)

Additional Sources

* **OpenAI API** – Real -time fraud detection using AI(Public, dynamic)
* **Kaggle Datasets** – E.g., “Credict card fraud,” (Public, static)

High-Level Methodology

Data Collection: The data will be gathered from banks or public sources having labeled transactional data(fraudulent and legitimate transactions).

Data Preprocessing: Missing value handling using imputation methods. Numerical features normalization to uniformize data distribution. Conversion of categorical variables into numerical variables using one-hot encoding or label encoding. Removal of outliers using statistical methods such as the IQR method.

Feature Engineering: Creating new useful features like transaction velocity, spending habit, and geospatial trends. Applying dimensionality reduction methods like PCA to improve the performance of models.

Model Training: Dividing the dataset into training and testing sets via stratified sampling. Training various machine learning models (Random Forest, Gradient Boosting, Voting Classifier) on pre-processed data. Hyperparameter tuning using methods like Grid Search or Bayesian Optimization.

Fraud Detection Algorithm: Developing an ensemble learning framework in which, Random Forest detects transaction anomalies based on decision trees. Gradient Boosting strengthens weak learners to enhance fraud detection. Voting Classifier combines model predictions for better classification accuracy.Utilizing a threshold-based classification system toreduce false positives.

Model Evaluation and Validation: Model evaluation with metrics like Precision, Recall, F1-score, and AUC-ROC. Cross-validation to ensure model generalizability. Comparative analysis with existing fraud detection models.

Deployment and Real-Time Monitoring: Deploying the trained model in a cloud or on-premise setup. Having an automated alert system for risky transactions. Regularly training the model on fresh transactional datato learn changing fraud patterns.The algorithm and methodology are designed to give a scalable, high-performance, and robust frauddetection system that increases the security offinancial transactions while reducing financial losses.This method uses the latest machine learning methods and real-time processing features to guarantee sound fraud detection in a constantly changing threat environment

Tools and Technologies

Programming Language :

Python :Chosen for its simplicity, readability, and wide range of libraries for data analysis and machine learning.

Google Colab :Cloud-based, free to use, supports GPU acceleration, and allows easy collaboration.

Alternatively, you can use Jupyter Notebook or VS Code if working locally.

Libraries :

* pandas – For data manipulation and analysis.
* numpy – For numerical operations.
* datetime – To handle and process date-time features.

Data Visualization :

* matplotlib – For creating basic plots (line, bar, scatter, etc.).
* seaborn – For enhanced statistical visualizations like heatmaps, boxplots, etc.
* plotly – For interactive and dynamic visualizations

Machine Learning & Modeling :

* scikit-learn – For preprocessing, regression models, and evaluation metrics.
* xgboost / lightgbm – For advanced gradient boosting models with high performance.
* statsmodels – For basic statistical modeling (optional).
* keras / tensorflow / pytorch – If using deep learning models like LSTM (optional for time series forecasting).

Optional Tools for Deployment :

* Streamlit – For building a quick and interactive web app to display results.
* Flask – Lightweight web framework to deploy models as APIs.
* Gradio – To create simple user interfaces for ML models (optional).
* FastAPI – For building high-performance APIs (optional, more advanced).

\*Machine Learning Models & Ensemble Techniques

\*Imbalanced Data Handling Techniques

Team Members and Roles

|  |  |  |
| --- | --- | --- |
| **Name** | **Role** | **Responsibilities** |
| ELUMALAI R | **Project Lead** | Oversee project development, coordinate team activities, ensure timely delivery of milestones, and contribute to documentation and final presentation. |
| MADHU MADHI B | **Data Engineer** | Collect data from APIs (e.g., Twitter), manage dataset storage, clean and preprocess text data, and ensure quality of input data. |
| DIVYA V | **NLP Specialist / Data Scientist** | Build sentiment and emotion classification models, perform feature engineering, and evaluate model performance using suitable metrics. |
| SUNDAR S | **Data Analyst / Visualization Lead** | Conduct exploratory data analysis (EDA), generate insights, and develop visualizations such as word clouds, emotion trends, and sentiment dashboards. |