DELIVERING PERSONALIZED MOVIE RECOMMENDATIONS WITH AI-DRIVEN MATCHING SYSTEM

STUDENT NAME: P.KALAIARASI

***REGISTER NUMBER*:** 412323106015

INSTITUTION: SRI RAMANUJAR ENGINEERING COLLEGE

***DEPARTMENT*:** BE. ECE

DATE OF SUBMISSION: 10/05/2025

GITHUB LINK : https://github.com/kalaiarasi 2. //phase

PROBLEM STATEMENT :

Credit card fraud is a growing concern, with significant financial losses for both cardholders and financial institutions. Traditional rule-based systems for detecting credit card fraud often struggle to keep pace with evolving tactics employed by fraudsters. These systems can generate high false-positive rates, leading to customer inconvenience and increased operational costs.

Challenges

1. *Sophisticated fraud patterns*: Fraudsters continually adapt and refine their methods, making it difficult for traditional systems to detect new patterns.

2. *High transaction volume*: Financial institutions process millions of transactions daily, making manual review impractical.

OBJECTIVES OF PROJECT:

PRIMARY OBJECTIVES

1. *DETECT AND PREVENT CREDIT CARD FRAUD*: UTILIZE AI-POWERED *SOLUTIONS TO IDENTIFY AND PREVENT FRAUDULENT TRANSACTIONS IN REAL-TIME*.

*2. *Minimize Financial Losses*: Reduce financial losses resulting from credit card fraud for both cardholders and financial institutions.*

*3. *Improve Customer Experience*: Enhance customer satisfaction by reducing false positives and ensuring legitimate transactions are processed smoothly.*

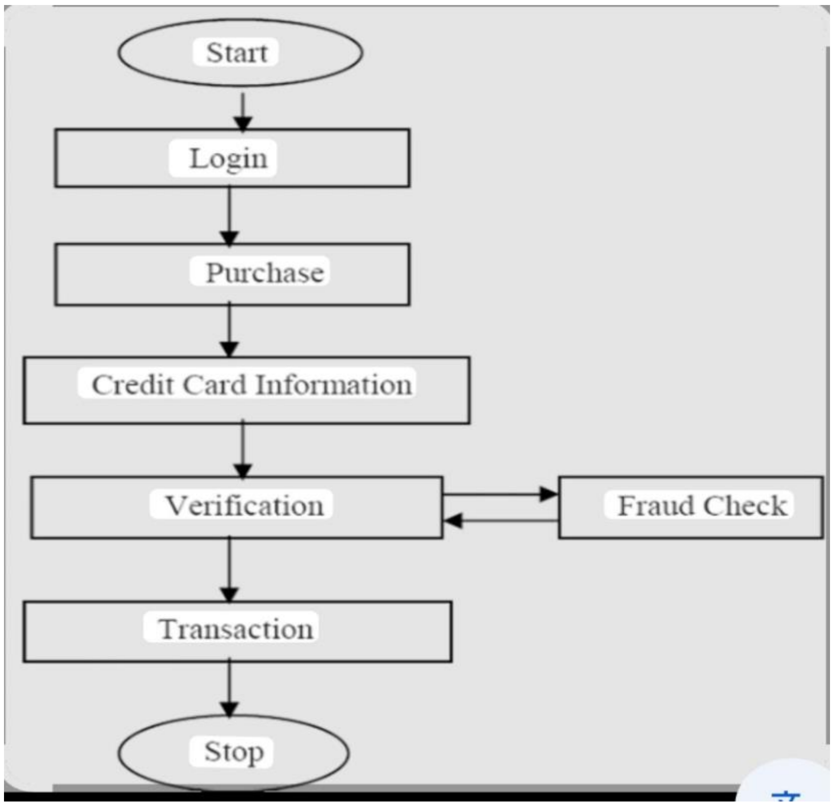
Secondary Objectives

*1. *Develop Advanced Analytics*: Leverage machine learning and deep learning techniques to analyze transaction data and identify complex patterns.*

*2. *Implement Real-time Processing*: Analyze transactions in real-time to detect and prevent fraud as it occurs.*

*3. *Continuously Update Models*: Regularly update models with new data to stay ahead of emerging fraud patterns.*

3. **Flowchart of the Project Workflow**



4. DATA DESCRIPTION

1. Data Overview:

The dataset used for credit card fraud detection typically contains transactional data, including both fraudulent and non-fraudulent transactions.

Key Characteristics:

Binary Classification Problem: Fraud (1) vs. Non-Fraud (0).

Imbalanced Data: Fraudulent transactions usually make up less than 1% of the data.

Anonymized Features: Due to privacy concerns, features might be encoded or anonymized.

Target Variable:

The typical target variable in credit card fraud detection is: is_fraud or fraud_flag:

This variable indicates whether a transaction is fraudulent or not.

It usually takes binary values:

1 = Fraudulent Transaction

0 = Non-Fraudulent Transaction

Static or Dynamic:

The target variable is_fraud is considered dynamic because:

It changes with every transaction.

Fraud patterns evolve over time, influenced by:

New fraud techniques.

Changes in user behavior.

Security updates in payment systems.

5. Data Preprocessing

1. Data Collection

Gather transaction data from diverse sources, including:

SPD Technology Transaction Logs: Details of each transaction (e.g., amount, timestamp).Customer Information: Demographics, account details.Device and Location Data: IP addresses, device IDs, geolocation.Historical Fraud Records: Previously identified fraudulent transactions.SPD Technology This aggregated data forms the foundation for training and evaluating your ANN model.

2. Data Cleaning

Ensure data quality by addressing common issues:Missing Values: Impute or remove records with missing entries.Inconsistencies: Standardize formats (e.g., date formats, categorical labels).Duplicates: Identify and remove duplicate records to prevent bias.Clean data ensures that the model learns accurate patterns without noise.

3. Feature Engineering

Enhance the dataset by creating informative features:Time-Based Features: Time of day, day of the week, time since last transaction.Transaction Amount Statistics: Rolling averages, standard deviations.Behavioral Patterns: Frequency of transactions per merchant, average spend.Device and Location Indicators: Changes in device usage or location anomalies.

SPD Technology :These engineered features help the ANN model capture complex patterns indicative of fraudulent behavior.

4.Handling Imbalanced Data

Address the common issue of class imbalance (i.e., fewer fraudulent transactions) using techniques such as:

Oversampling: Duplicate instances of the minority class.

Undersampling: Reduce instances of the majority class.

Synthetic Minority Over-sampling Technique (SMOTE): Generate synthetic examples of the minority class to balance the dataset.

Balancing the dataset prevents the model from being biased toward the majority class.

5. Data Splitting

Divide the dataset into training and testing subsets

Training Set: Typically 70-80% of the data, used to train the model.

Testing Set: The remaining 20-30%, used to evaluate model performance.

Ensure that the split maintains the class distribution (stratified sampling) to provide an accurate assessment of model performance.

6 .Model Training and Evaluation

.6. Exploratory Data Analysis (EDA)

1. Class Distribution

The dataset is highly imbalanced, with fraudulent transactions constituting only 0.172% of all transactions. This imbalance poses challenges for model training, as models may become biased toward the majority class.

2. Statistical Summary

Analyzing the statistical properties of the features helps in understanding the data distribution:

Time:Range: 0 to 172,792 seconds.

Observation: No significant difference in the distribution of 'Time' between fraudulent and legitimate transactions.

Towards AI

Amount:Range: 0 to 25,691.16.

Observation: Fraudulent transactions tend to have lower amounts compared to legitimate ones.

GitHub

+10

Towards AI

+10

GitHub

+10

4. Feature Correlation

Since features V1 to V28 are the result of PCA transformation, they are uncorrelated by design. However, analyzing the correlation between 'Amount' and 'Class' can provide insights:

Correlation between 'Amount' and 'Class': Approximately -0.01, indicating a negligible negative correlation.

5. Visualization

Visual tools can uncover patterns not evident in statistical summaries:

Histogram of 'Amount':

Legitimate transactions exhibit a wide range of amounts.Fraudulent transactions are more concentrated in lower amounts.Boxplot of 'Amount' by 'Class':

Highlights the difference in transaction amounts between fraudulent and legitimate transactions.

Time vs. Amount Scatter Plot:

No discernible pattern observed in the timing of fraudulent transactions.

6. Anomaly Detection

Identifying anomalies can aid in detecting fraudulent transactions:

Isolation Forest:

An unsupervised learning algorithm effective in detecting anomalies in high-dimensional datasets.

Local Outlier Factor (LOF):

Measures the local deviation of a data point with respect to its neighbors, useful for identifying outliers.

7. Dimensionality Reduction

Reducing the dataset's dimensionality can help in visualization and model performance:

t-SNE (t-Distributed Stochastic Neighbor Embedding):

Effective in visualizing high-dimensional data by reducing it to two or three dimensions.

PCA (Principal Component Analysis):

Already applied in the dataset, resulting in features V1 to V28.

GitHub

+5

GitHub

+5

GitHub

+5

8. Handling Class Imbalance

Addressing class imbalance is crucial for model accuracy:

SMOTE (Synthetic Minority Over-sampling Technique):

Generates synthetic samples of the minority class to balance the dataset.

Undersampling:

Reduces the number of samples in the majority class to balance the dataset.

9. Feature Importance

Determining which features contribute most to fraud detection:

Towards AI

+5

GitHub

+5

Geekster

+5

By conducting thorough EDA, we gain valuable insights into the dataset's structure and the characteristics of fraudulent transactions. These insights inform the development of more accurate and robust AI models for credit card fraud detection and prevention.

7. **Model Building**

Data Preprocessing:

Load and explore the data.Handle class imbalance with SMOTE.

Scale numerical features.

Split data into training and testing sets.

Model Selection and Training:

Use classification algorithms like Logistic Regression, Random Forest, or XGBoost.

Train multiple models for comparison.

Model Evaluation

Metrics: Accuracy, Precision, Recall, F1-score, AUC-ROC.

Plot Confusion Matrix and ROC Curve.

Model Deployment:

Save the trained model for real-time inference.

Integrate with a real-time transaction monitoring system.

8. Visualization of results and model insights

Model Performance:

Confusion Matrix

ROC CCurve

Precision-Recall Curve

Data Insights:

Fraud vs. Non-Fraud Distribution

Feature Importance

Correlation Heatmap

Insights and Interpretation

Confusion Matrix:

True Positive (TP): Fraud correctly detected.

True Negative (TN): Non-fraud correctly identified.

False Positive (FP): Non-fraud classified as fraud (false alarm).

False Negative (FN): Fraud missed by the model (risky).

ROC Curve:

AUC (Area Under Curve) closer to 1 indicates a strong model.

Shows the trade-off between Sensitivity (Recall) and 1 - Specificity.

Precision-Recall Curve:

Useful for imbalanced data.

High precision with high recall indicates a robust model.

Feature Importance:

Identifies which features are most influential in detecting fraud.

Helps with feature selection and model optimization.

Model Comparison:

Shows the best-performing model based on Accuracy, Precision, Recall, and F1-score.

Helps decide which model to deploy.

9.Tools and technology

Programming language : python

IDLE /NOTEBOOK : google colab, jupyter notebook etc..

Libraries : pandas, numpy, seaborn, matplot ,dataset.csv , skilearn, etc.

10.TEAM MEMBERS AND ROLES:

KALAIARASI -- : Problem formulation, data preprocessing, and feature engineering

PAVITHRA -- Collaborative filtering model (SVD), evaluation metrics, and hybrid

Integration

S. NITHYA SRI– Exploratory Data Analysis (EDA), visualizations, and

contentbased filtering model, Report writing, documentation, and GitHub

integration