CREDIT CARD FRAUD DETECTION

PYTHON FILE

Abstract

Algorithm for machine learning-based system that analyzes transaction data in real-time, effectively detecting credit card fraud while minimizing false positives.

Developing a real-time credit card fraud detection system involves several steps, including data pre-processing, model training, and continuous monitoring. Here's a high-level algorithm to guide the development of such a system:

Import necessary libraries and modules:

```
import pandas as pd

import numpy as np

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import StandardScaler

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, roc_auc_score
```

Step 1: Data Collection

- Obtain a real-time stream of credit card transaction data

```
while True:
```

Step 2: Data Pre-processing

- Pre-process the incoming transaction data
- Handle missing values, outliers, and data quality issues
- Normalize numeric features (e.g., transaction amount) in real-time

Step 3: Feature Engineering

- Calculate features such as transaction frequency, amount deviations, and time-based features in real-time

Step 4: Model Selection

- Select a pre-trained model (e.g., Random Forest) or train a model in real-time

Step 5: Model Training

- Train the selected model on historical data (if applicable)

Step 6: Real-time Prediction

- Use the trained model to make real-time predictions for incoming transactions
- Set a decision threshold to classify transactions as either fraud or legitimate

Step 7: Monitoring and Alerting

- Continuously monitor the system's performance
- Send alerts in real-time when suspicious transactions are detected

Step 8: Evaluation

- Calculate evaluation metrics for the real-time predictions
- Metrics can include accuracy, precision, recall, F1-score, and ROC-AUC

Step 9: Adaptive Learning (Optional)

- Periodically retrain the model with the latest data to adapt to changing fraud patterns

Step 10: Logging and Reporting

- Log all transactions and predictions for auditing and reporting purposes

Step 11: User Feedback (Optional)

- Collect feedback from users and fraud analysts to improve the system

Step 12: Compliance and Security

- Ensure the system complies with data privacy and security regulations

Step 13: Continuously iterate and improve the system

Step 14: End of Transaction Processing Loop

A real-time credit card fraud detection system is outlined in this method in a basic manner. Each stage could, in actuality, include more intricate considerations and sub-processes. As well as being difficult and requiring careful design and execution, choosing the right machine learning model, setting thresholds, and managing real-time data streams can all be problematic.

As fraudulent transactions are normally uncommon, the system should also include procedures for addressing class imbalance, setting warning thresholds, and guaranteeing data privacy and security. For the system to be successful and flexible enough to respond to changing fraud trends, continual monitoring, upkeep, and feedback loops are also necessary.