**Credit Fraud Detection Using the Hidden Markov Model**

Team Members:

Sergio Gabriel Jiawei Kun ; s\_kun@mail.fhsu.edu

Lee Joseph Judkins ; <insert your fhsu email>

Project Github: https://github.com/kunsergio117/CreditFraudDetectionHMM.git

Instructor: Prof. Hieu Vu

**Revision History:**

|  |  |
| --- | --- |
| Part 1 | 23/09/2024 |

# Individual Contributions Breakdown

Both Members contributed equally in both the front-end and HMM algorithm.

Table of Contents

[Individual Contributions Breakdown 2](#_Toc177773486)

[Section 1:  Customer Problem Statement 4](#_Toc177773487)

[Section 2:  System Requirements 5](#_Toc177773488)

[Project Management 6](#_Toc177773489)

[References 7](#_Toc177773490)

# Section 1:  Customer Problem Statement

Credit fraud remains a substantial concern for financial institutions worldwide, posing significant risks not only to the banks but also to their customers. The primary challenge lies in the sophisticated techniques employed by fraudsters, who continuously innovate and adapt their methods to exploit vulnerabilities in existing detection systems. Current fraud detection models predominantly rely on static, rule-based algorithms that offer limited flexibility and adaptability, leading to several critical issues.

One of the most pressing problems is the **High False Positive Rate** associated with traditional fraud detection systems. These systems often flag legitimate transactions as fraudulent due to their reliance on generalized pattern recognition and heuristics. For instance, when a customer makes a sizable purchase or travels internationally, the system might automatically trigger a fraud alert, causing unnecessary inconvenience. This not only frustrates customers—who may find their accounts unexpectedly frozen—but also inundates financial institutions with an overwhelming volume of false alarms, complicating the investigative process and straining their resources.

Another significant issue is the **Inability to Adapt** rapidly to evolving fraudulent techniques. Fraudsters are agile and continually refine their tactics, exploiting the inflexibility of older systems that require manual intervention for updates. Rule-based systems typically lack the ability to learn from new data; they operate reactively rather than proactively. This failure to keep pace results in many fraud incidents going undetected until significant damage occurs, often leaving banks scrambling to compensate for losses and restore customer trust.

**Data Volume** presents an ongoing challenge in fraud detection as well. Financial transactions are conducted at an unprecedented scale, with millions of credit card transactions processed daily in digital and physical environments. Conventional systems struggle with the sheer volume of data, often leading to delays in fraud detection and response times. In real-time transaction scenarios, such as those occurring at a retail point of sale or during online purchases, the inability to analyze this vast quantity of data dynamically can mean the difference between stopping fraud in its tracks and allowing a substantial loss to occur.

Existing approaches often lack a granular understanding of individual behavior, struggling to recognize deviations from typical patterns. For example, a customer who occasionally makes large purchases as part of their regular spending behavior might be mischaracterized as a fraud risk, resulting in missed opportunities for legitimate transactions. Consequently, there's a significant need for a solution that can accurately model normal transaction behaviors and effectively identify anomalous patterns.

The scenario is further complicated by historical data limitations; many existing systems are built on fixed rules applied to historical data that may not account for evolving consumer behaviors and emerging fraud tactics. As a result, fraudulent transactions slip through undetected, leading to increased losses for banks and decreased confidence among consumers regarding the safety of their financial transactions.

Our project aims to develop a comprehensive fraud detection system utilizing **Hidden Markov Models (HMM)**. The proposed HMM-based system will model the sequential nature of credit card transactions, effectively capturing the temporal patterns that characterize normal customer behavior. Through this approach, the system will be better equipped to recognize deviations indicative of fraudulent activity, resulting in enhanced detection accuracy, reduced false positive rates, and an overall more responsive system that works in real-time.

The

By allowing the system to learn from data over time, it can dynamically adapt to evolving fraud schemes. Furthermore, offering insights and reporting tools will empower analysts to investigate flagged transactions efficiently and accurately, ultimately bolstering confidence in the financial institutions’ fraud detection capabilities and improving overall customer satisfaction.

This robust and adaptable fraud detection system will enrich the landscape of financial security, ensuring that customers can engage in transactions with greater peace of mind, while financial institutions can mitigate their losses effectively and efficiently.

# Section 2:  System Requirements

## Functional Requirements:

| **Identifier** | **Requirement** |
| --- | --- |
| **REQ1** | The system shall allow the user to upload transaction data in CSV format for analysis. |
| **REQ2** | The system shall provide a user interface with options to visualize transaction patterns over time, categorized by various parameters such as amount, date, and merchant. |
| **REQ3** | The system shall enable users to set thresholds for alerts regarding fraudulent transactions based on statistical anomalies detected by the Hidden Markov Model. |
| **REQ4** | The system shall include an API to fetch real-time transaction data and historical data from external sources to enhance model accuracy. |
| **REQ5** | The system shall have a reporting feature where users can generate and download reports summarizing detected fraudulent activities and system performance. |
| **REQ6** | The system shall include a user-friendly dashboard displaying key metrics such as false positive rates, detection accuracy, and other relevant KPIs. |
| **REQ7** | The system shall facilitate user authentication to ensure secure access to the application and protect sensitive financial data. |

## Non-Functional Requirements:

| **Identifier** | **Requirement** |
| --- | --- |
| **REQ8** | The system shall be designed with a user-friendly interface that minimizes clutter and promotes ease of navigation. |
| **REQ9** | The system shall maintain high availability, ensuring that the service is operational at least 99.5% of the time to cater to users in need of real-time fraud detection. |
| **REQ10** | The system shall provide mobile compatibility, allowing users to access features on smartphones and tablets without losing functionality. |
| **REQ11** | The system shall utilize efficient algorithms to provide timely alerts regarding potential fraudulent activities with minimal latency. |
| **REQ12** | The system shall ensure data privacy and compliance with relevant regulations (e.g., GDPR, PCI DSS) regarding user data handling and storage. *(This is a regulation for financial data use according to our research and I am including this for the sake of treating this as a real shareholder pitch; Question: should we remove this requirement?)* |
| **REQ13** | The system shall provide an admin interface for monitoring system performance, managing user accounts, and configuring detection parameters. |

## On-Screen Appearance Requirements:

| **Identifier** | **Requirement** |
| --- | --- |
| **REQ14** | The system shall visually represent data using dynamic charts and graphs to illustrate patterns and anomalous transactions effectively. |
| **REQ15** | The system shall maintain a consistent and simple graphical user interface (GUI) across both web and mobile applications, providing an intuitive user experience. |

### FURPS Table:

| **Category** | **Description** |
| --- | --- |
| **Functionality** | The system will leverage Hidden Markov Models for detecting fraudulent credit card transactions, providing features such as transaction uploads and real-time monitoring of anomalies. |
| **Usability** | The system will include a straightforward main page with intuitive navigation to all key functionalities, helping users access important features quickly. All input fields will be clearly labelled for improved user experience. |
| **Reliability** | The system will perform input validation to prevent user errors when uploading transaction data. Version control will manage application updates effectively to prevent data loss and maintain integrity. |
| **Performance** | The application will support multiple users concurrently with minimal latency, ensuring that calculations and alerts are generated efficiently to maintain real-time detection capabilities. |
| **Supportability** | The system will be compatible across popular web browsers. The code structure will be modular, allowing for easy updates and feature expansions. |

# Project Management

Both team members will be actively engaged in the development process, utilizing the GitHub repository to track progress and contributions clearly.

Responsibilities will be delegated based on each member's strengths and expertise, with a shared accountability structure ensuring that all aspects of the project are covered. This is because our group consists of only a pair and thus realistically we will both need input and validation from one another in all development areas.

# References